

5-2018

## A Novel Magnetorheological Fluid Damper for a Small Spacecraft with Flexible Appendages

Robert Waelchli

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Space Vehicles Commons](#), and the [Structures and Materials Commons](#)

---

### Scholarly Commons Citation

Waelchli, Robert, "A Novel Magnetorheological Fluid Damper for a Small Spacecraft with Flexible Appendages" (2018). *Dissertations and Theses*. 391.  
<https://commons.erau.edu/edt/391>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact [commons@erau.edu](mailto:commons@erau.edu).

A NOVEL MAGNETORHEOLOGICAL FLUID DAMPER FOR A SMALL  
SPACECRAFT WITH FLEXIBLE APPENDAGES

By

Robert Waelchli

A thesis submitted in partial fulfillment of the requirements for the degree of  
Master of Science in Mechanical Engineering

Embry-Riddle Aeronautical University

Daytona Beach, Florida

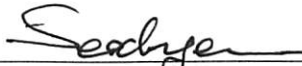
May, 2018

A NOVEL MAGNETORHEOLOGICAL FLUID DAMPER FOR A SMALL  
SPACECRAFT WITH FLEXIBLE APPENDAGES

By

Robert Waelchli

THESIS COMMITTEE:



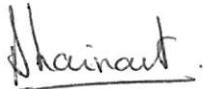
Dongeun Seo, Ph.D.  
*Committee Chair*



Daewon Kim, Ph.D.  
*Committee Member*



Darris White, Ph.D.  
*Committee Member*



Jean-Michel Dhainaut, Ph.D.  
*Mechanical Engineering, Master's Program Coordinator*

4/16/18  
Date



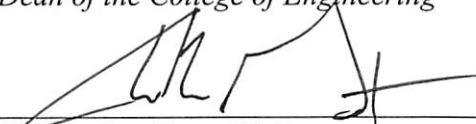
Eduardo Divo, Ph.D.  
*Mechanical Engineering, Department Chair*

4/16/18  
Date



Maj Mirmirani, Ph.D.  
*Dean of the College of Engineering*

4/17/18  
Date



Christopher Grant, Ph.D.  
*Vice Chancellor, Academic Support*

4/17/18  
Date

## ACKNOWLEDGEMENTS

Particular acknowledgement is due the following people who were instrumental in completing specific portions of my research: Dr. Jeffrey Kauffman and Garrett Lopp, University of Central Florida, who provided lab equipment and expertise with the impact hammer modal testing; Dr. Claudia Moreno, who provided equipment and expertise with the shaker table vibration testing and post-processing of the data; and Dr. David Sypeck who provided lab equipment and guidance during early portions of the research that allowed for some preliminary experiments.

Special thanks is also due to Dr. Jean-Michel Dhainaut, who coordinated my graduate program and who made it possible for me to complete my research on-time and with no administrative burdens. I appreciate his efforts.

My work would not have been possible without the assistance of my committee members, Dr. Daewon Kim and Dr. Darris White. They provided materials and lab space of course, but more than that, they provided ideas and instruction. Their knowledge and their generosity have had a lasting impact on me.

Most important to my education overall has been the efforts of my thesis advisor, Dr. Dongeun Seo. He has been my guide to a world of higher thinking—math, physics, and philosophy—and I am the better for it. I am deeply grateful.

Finally, I would like to acknowledge my family whose unfailing support and love have always been the foundation for all my success.

## TABLE OF CONTENTS

Thesis Review Committee .....	i
Acknowledgements .....	ii
Table of contents .....	iii
List of Figures .....	iv
Abstract .....	1
I. Introduction .....	2
A. Background .....	2
B. Problem Statement and Research Question .....	6
II. Literature Review .....	7
A. Current Challenges of Small Satellites.....	7
B. Magnetorheological (MR) Fluids.....	10
III. Methodology.....	13
A. System Description .....	14
B. Numerical Model.....	19
C. Damping Matrix Definition.....	25
1. Impact Hammer Modal Testing.....	25
2. Shaker Table Vibration Testing.....	32
D. Controller Implementation .....	36
IV. Results .....	38
A. Impact Hammer Modal Testing .....	38
B. Shaker Table Vibration Testing .....	45
C. Completed System Description and Performance.....	49
V. Conclusions and future work.....	52
References.....	55
Appendix A – Additional Curve-Fitting Plots .....	A-1
Appendix B – Numerical Model MATLAB <sup>TM</sup> Script .....	B-1

## LIST OF FIGURES

Figure 1. Right-side view of the initial satellite-damper system configuration.....	15
Figure 2. Top-down view of the initial satellite-damper system configuration.....	15
Figure 3. An illustration of one alternative spacecraft-damper system configuration.....	16
Figure 4. Profile view of a third spacecraft-damper configuration.....	17
Figure 5. Body-fixed coordinate system (BFF) .....	19
Figure 6. Relationship between the BFF and LVLH coordinate systems.....	20
Figure 7. First test article .....	26
Figure 8. Internal flow paths of the second test article .....	26
Figure 9. Second test article.....	27
Figure 10. Third test article.....	27
Figure 11. Custom-built test mount .....	28
Figure 12. Second test article, in the test mount, with illustrated field lines and poles....	29
Figure 13. Demonstrator tube used to measure the third test article's field strength .....	30
Figure 14. Half-power points of a resonant peak.....	31
Figure 15. Fourth test article, used for shaker table vibration testing.....	32
Figure 16. The fourth test article, mounted during shaker table vibration testing.....	33
Figure 17. Bode plot—Estimated versus actual TF in the vicinity of the first mode resonant peak .....	35
Figure 18. FRF resulting from impact hammer modal testing of the first test article .....	38
Figure 19. Accelerometer mounting locations.....	39
Figure 20. FRF of first test article with external 25-30mT field .....	40
Figure 21. Bode plot—First test article, first mode resonant peak shift .....	40
Figure 22. FRF resulting from impact hammer modal testing of the second test article..	41
Figure 23. Bode plot—Second test article, first mode resonant peak.....	42
Figure 24. FRF resulting from impact hammer modal testing of the third test article .....	43
Figure 25. Bode plot—Third test article, first mode resonant peak shift, 0 to 5A .....	44
Figure 26. Bode plot—Third test article, first mode resonant peak shift, 5A to 10A.....	44
Figure 27. Frequency sweep of 5-40Hz with the fourth test article; no magnetic field ...	46
Figure 28. Experimental setup for the shaker table vibration testing .....	46
Figure 29. Frequency sweep of 5-40Hz with the fourth test article; 25-30mT applied magnetic field.....	47
Figure 30. Simulation results from a two-axis slewing maneuver.....	49
Figure 31. Two-axis slewing maneuver with three different field strengths .....	50

## ABSTRACT

Small satellites have become increasingly popular over the past thirty years, particularly since the adoption of the common CubeSat architecture early this century. Because of their restricted volume and electrical budgets however, there are practical limits to the missions that small satellites may adopt. One potential near-term solution to the problem of limited electrical power may be the adoption of larger, flexible solar arrays.

However, spacecraft with flexible appendages have historically presented attitude control challenges relating to platform stability given the dynamic response of the flexible components to applied torques. These challenges may be particularly disruptive to a small spacecraft with low inertia. Previous studies have examined minimizing the dynamic motion of flexible appendages via shaping control of the external torquers (attitude actuators), and damping the dynamic responses in various schemes.

This thesis presents the possible design of a new damper for the small, flexible spacecraft model. The design takes advantage of a smart material known as a magnetorheological (MR) fluid that was initially created in the 1940s, but has been subject to renewed interest over the past 30 years. A numerical model of the damper system is described and test articles with representative properties are subjected to dynamic testing to inform the model behavior. Completed simulation results for a spacecraft slewing maneuver are presented along with suggestions for future research and future design iterations.

## **I. INTRODUCTION**

### **A. Background**

Small satellites, meaning those with a mass of 1-100kg, are becoming increasingly common. Between 2000 and 2010, fewer than 180 were placed into orbit, however since 2011, more than 750 have been [1, 2]. Moreover, beyond simply increased numbers, small satellite mission profiles have also grown in complexity. Prior to 2018, all missions flown within this class of satellite were restricted to Earth orbit, but in May of 2018, NASA intends to launch two small satellites along with their Mars Insight probe. NASA's Jet Propulsion Laboratory intends to use these satellites—they're formally named Mars Cube One (MarCO)—as communication relays during the “entry, descent, and landing” phase of the Insight mission [3]. Additional small satellite missions are planned for the Moon and nearby asteroids [4, 5]. Many of these small satellites, including the beyond-Earth-orbit examples presented here, take advantage of a relatively new satellite architecture, the CubeSat.

The CubeSat model is a common satellite standard that describes a specific shape, weight, and size. Since its inception, this standard has resulted in mass-produced components and hardware that allow for relatively inexpensive satellite production [6]. The CubeSat idea was originally conceived by Bob Twiggs (then of Stanford) and Dr. Jodi Puig-Suari (Cal Poly San Luis Obispo) and presented at the 14<sup>th</sup> Annual Utah State University Conference on Small Satellites [7]. Their original paper described the CubeSat as a 10cm cube with the same mass as a similar cube of water, 1kg. Since that original paper, which focused largely on the single-unit CubeSat, three-unit CubeSats (10cm x



10cm x 30cm) have become the dominant model comprising more than two-thirds of the more-than-600 CubeSats launched since 2013 [8].

While popular, all small satellites are, by definition, limited in at least two areas. First, there is a fixed, small volume that must be shared between the research payload and the equipment necessary for mandatory spacecraft functions (attitude control, thermal control, power, etc.). Second, there is limited electrical power available for operations. Ultimately, the continued evolution of electronics into smaller and less power-consuming packages may make both limitations irrelevant, however a reasonable near-term solution to the power problem could include the adoption of larger, flexible solar arrays. But, even as flexible solar arrays could solve one problem, they introduce another; spacecraft attitude control when rigid-body-motion can no longer be assumed.

The operation of spacecraft with flexible appendages has been a challenging attitude control problem since the launch of America's first satellite, Explorer 1. That vehicle quickly adopted a persistent off-nominal orientation due to the non-rigid-body motion of its four flexible antennae [9]. For all space vehicles with flexible components, the vibratory motion of dynamical systems operating in vacuum is particularly troublesome because the response to input torques may be highly non-linear, long-lived, and governed in part by complex interactions between modes [10, 11, 12]. The destabilizing effects of these dynamic responses are proportional to the size and mass of the appendages themselves and the vibration amplitude, but they may be significant enough to disrupt mission goals. [11]. The vibration of a flexible spacecraft can be particularly disruptive for missions with precise pointing requirements such as

communications relays for deep space missions [11], or observation spacecraft with narrow targeting windows requiring a high degree of stability [13].

The dynamic attitude control and vibration damping problem for flexible spacecraft has been studied in detail since at least 1970 [13], and published research is available on a large number of mitigation strategies. These studies mostly focus on one of two paths or, frequently, a combination of the two. First, structural vibration is minimized through judicious actuation of traditional spacecraft torquers (thrusters, reaction wheels) achieved by various controller architectures rather than simple bang-bang control logic. Examples include Positive Position Feedback (PPF) [14], Sliding Mode Control (SMC) [15],  $H_\infty$  [16], Linear Quadratic Regulator (LQR) [17], and many others including hybrids. Second, either alternatively or in tandem, dynamic responses are minimized with active or passive vibration-suppression employing smart materials, most commonly piezoelectric sensors and actuators. [10, 11, 14, 18] Although some of these solutions are adaptable to the small satellite model, many are not due to size and power restrictions. In any event, with the popularity of small satellites increasing, new ideas and further advances in flexible-satellite attitude control, in particular solutions adaptable to the small-satellite model, will enable further growth in this area.

In the last twenty years, a smart material first developed in the 1940s has found new commercial applications and has consequently become the subject of reinvigorated study in a number of fields. Magnetorheological (MR) fluid, or MRF, is a material in which micro-sized ferromagnetic particles are suspended in some type of oil or viscous “carrier” fluid [19]. Additional additives, surfactants, may further be added to minimize settling [20]. In the absence of a magnetic field, the mixture flows like other Newtonian

fluids and will adopt the shape of its container, but when a magnetic field is applied, a yield strength rapidly develops as the ferromagnetic particles align themselves in chain-like structures along the lines of magnetic force [19, 20]. The material properties of the semi-solid are proportional to the strength of the applied magnetic field. The response time of the fluid to the applied magnetic field is quite short, typically several milliseconds, leading to an extremely responsive, adaptable system.

For the small satellite model, the benefits of an MR fluid-based damping system would be three-fold. First, simple MR fluid systems can be relatively cheap and easy to construct. Second, because they can be used as an active damping system and provide a relatively large damping force [21], they have the potential to provide excellent damping properties. Finally, third, experience with automobile dampers shows that the power required to provide an effective damping response is quite low, typically less than 10 watts [21], thus potentially enabling a high degree of active damping capability at modest cost.

## **B. Problem Statement and Research Question**

This thesis approaches the current small satellite environment from the perspective that as small satellites become increasingly capable and are increasingly adopted for demanding mission profiles, the need for larger, flexible solar arrays will become more pronounced. That possibility will require solutions to the dynamic attitude control problems introduced with flexible appendages. More formally, this problem statement is presented as follows:

**PS:** *Flexible solar arrays could be a practical near-term solution for increased power to small satellites, but they complicate attitude control relative to rigid-body appendages.*

With the goal of providing solutions to the problem thus posed, this thesis investigates whether an MR fluid-based damper could provide effective damping for the small satellite model. To appropriately restrict the scope of the effort, the research question this thesis attempts to answer is formally stated as shown:

**RQ:** *What damping performance, in terms of settling time and modal damping ratios, could be expected from a magnetorheological fluid damper when used for a small spacecraft with flexible solar arrays?*

To provide an answer to that question, this thesis concentrates on several variations of a simple, prototypical MR fluid damper design. Utilizing a numerical model supplemented with experimental data, the possible effectiveness of the design is investigated. Further avenues for research are also presented in the final section.

## **II. LITERATURE REVIEW**

The investigation described in this thesis involves two active areas of academic study: 1) The evolution of small satellites, and the limitations they face in terms of power production for increasingly complex missions, and 2) MR fluids and their applications. Journals and conferences concerned with space flight mechanics and materials science contain active research in both areas, however to the best of the author's knowledge, no research has been conducted to date that combines them together as presented here.

Therefore, in order to accurately reflect the current context in which this research has been conducted, the reviewed literature is presented in each area individually. First, literature related to small satellites themselves, CubeSats in particular, and their changing mission roles is presented and then second, the literature relating to the creation and application of MR fluids from 1948 to the present.

### **A. Current Challenges of Small Satellites**

Dr. Siegfried Janson of the Aerospace Corporation writes in "Small Satellites, Past, Present, and Future" [22] that the generic term "small satellites" actually encompasses multiple, more specific categorizations defined by mass. The categories between 1-10 kg and 10-100kg, the sizes referenced in this thesis, are formally referred to as nanosatellites and microsatellites respectively. Janson writes that by the 1990's, microsatellites in particular were experiencing a resurgence at about 9 launches per year following a relative dearth of small satellites in the 1970's and 1980's. Dr. Martin Sweeting of IEEE and the Royal Society attributes the late-20<sup>th</sup> century resurgence in microsatellites to the evolution of commercially available microelectronics [23] and

writes that the new-found technology allowed for satellite construction under new management structures. No longer were satellites simply the domain of governments or large organizations, but smaller institutions with more agile, IT-like management structures were becoming involved.

It was in this environment that the CubeSat architecture was first presented at the 14<sup>th</sup> Annual Utah State University Conference on Small Satellites [7]. Dr. Jordi Puig-Suari (California Polytechnic, San Luis Obispo) and Professor Robert Twiggs (then of Stanford), presented CubeSat as a generic small satellite standard. The common form-factor, built around 10cm or “unit” cubes, had the benefit of allowing for standardized processing and launch infrastructure. Additionally, CubeSats could be built with mass-produced, off-the-shelf components that could reasonably be assembled by scientists, universities, high schools, or clubs; anyone with the will, but not necessarily the resources to construct larger, more traditional satellites. Since that initial proposal, there has been a wide-spread adoption of most elements of the model. Dr. Michael Swartwout, Chair of the Aerospace and Mechanical Engineering department at St. Louis University, maintains an extensive website devoted to tracking CubeSat mission histories. His data reflect that after a modest mission rate from 2000-2011, averaging about 8 satellites per year, there has since been a rapid increase, peaking with more than 280 CubeSats in 2017 [24]. These missions, mostly of the 3U form-factor comprising a rectangular satellite 10cm x 10cm x 30cm, have come to dominate the small satellite market comprising more than 90% of the 1-50kg space-vehicles orbited last year [24, 2].

Since 1999, hundreds of journal articles and conference papers have been published regarding specific CubeSat missions or the development of key enabling

technologies. Some of the most prominent research has been published since 2010 as missions have become more ambitious. Recent notable examples include several regarding the QB50 mission involving tens of CubeSats flying in formation with dozens of instruments [25, 26, 27]; multiple examples regarding various prototype solar sail demonstrations with potential interstellar applications [28, 29, 30, 31] and several additional examples concerning interplanetary destinations such as the moon and Mars [3, 32]. From the start in 1999, the power limitations of the CubeSat model were understood, but little published literature is available earlier than 2010 that indicates pronounced interest in overcoming that challenge.

As mission applications for CubeSats have become more ambitious, particularly in the last 8 years, researchers have begun investigating increased power budgets for the CubeSat model. Since volume is by definition limited, the vast majority of these ideas have focused on increasing the number of available solar cells with various types of deployable rigid solar panels [33, 34, 35, 36] or in one case, inflatable [37]. Additionally, at least three papers, one from 2012 [38] and two from 2015 [39, 40], deal with increasing the efficiency of components of the electrical distribution system.

The new research presented in this paper does not intend to supplant the important research into the areas of increasing efficiency or deployable rigid solar arrays. Rather the goal is to open the door to an additional solution that may further enable the continued evolution of the small spacecraft's capabilities. The author is unaware of current research into the use of flexible solar arrays of the type described here with the CubeSat model.

## **B. Magnetorheological (MR) Fluids**

MR fluids were first created in 1948 by Jacob Rabinow working for the National Bureau of Standards [41]. Rabinow published in November of that year, and in that first article he discussed multiple applications for the nascent technology including hydraulic actuators, dashpots, and clutches. He further described the fluid as holding “considerable promise” for additional applications beyond even those.

Dr. Roger Stanway, University of Sheffield, writes in his smart-fluids research survey [42] that despite its seemingly broad potential, for more than 30 years after its creation, the new technology found few commercial applications. However, in the mid 1990’s interest in MR fluid was re-awakened—most prominently due to new attention from the American automobile industry, but with applications in other areas as well. In 2002, there was a prominent commercial example when GM began producing their luxury Cadillac Seville model with MR fluid-based suspension struts [43]. MR fluid-based suspensions have since been adopted by Audi, Ferrari, and Holden Special Vehicles [44].

Research also continued for other automotive applications. Park, Stoikov, da Luz, and Suleman [45], Park, da Luz, and Suleman [46], and Karakoc, Park, and Suleman [47], all presented research investigating MRF for automotive braking applications although their final results indicate that they were unable to generate sufficient stopping torque. More recently, Yu, Ma, Song, and Liu [48] described research that leveraged the same technology, but with innovations to amplify the braking force. Less demanding braking applications—feedback resistance for exercise bikes—were successfully demonstrated as early as the mid-90’s and by 1995 the Nautilus Corporation began using MR fluid brakes in their exercise equipment [49, 50].



Another area of commercial use, unrelated to cars or braking, has been as a high-quality polishing surface for precision optics. Research in this area began in the mid-90's at the University of Rochester Institute of Optics and by 1998 the company QED Technologies had been founded and was selling their first commercial MRF polishing machine [51]. In their 2001 survey of current and potential MR-fluid applications, Wang and Meng [21] described three advantages of MR polishing. First, there is a high controllability of the grinding surface; second there is constant replenishment of the abrasive material—the fluid; and third, there is constant medium for heat and debris removal.

Beyond these existing commercial uses, MR fluid continues to be the subject of active academic research. In 2016, Seo and Lee proposed a novel attitude control scheme using pressurized flow through three MR-fluid filled rings [52]. They described a system in which the viscosity of the fluid was independently controlled in each ring. This created differential fluid velocities and hence produced a controllable torque on the spacecraft. The research was conducted as a numerical simulation, and to date a physical demonstration has not been performed.

Also relevant to the new research described in this paper is a 2015 experiment conducted by Kaluvan, Park, and Choi at Inha University in South Korea [53]. In that experiment, in which the goal was to create a tunable damper for terrestrial applications, Kaluvan and his colleagues used a flexible nylon tube twisted into a helical shape and surrounded by hundreds of loops of narrow-gage conductor. The tube was then filled with MR fluid and capped. While varying the current through the conductor, Kaluvan, Park, and Choi demonstrated that the material properties of the spring were controllable in a

series of applied force tests; the spring required about 125% more force, more than 900mN at 0.8A compared to about 650mN at 0A, to elongate the spring 3.5cm.

The author is unaware of research to date that investigates the performance of any design of MRF damper for small space vehicles with flexible components.

### III. METHODOLOGY

The research described in this thesis was conducted in three phases. For clarity, they are briefly discussed below followed by more detailed explanations.

In the first phase, the system model of a small satellite with flexible solar arrays was numerically constructed using MATLAB<sup>TM</sup>. The system's equations-of-motion were derived using a Lagrangian energy approach that resulted in a system of four coupled ordinary differential equations. The model includes a definition a stiffness matrix  $[K]$  and an inertial matrix  $[M]$  that follow naturally from the system's geometry and material properties. For completeness, a damping matrix was inserted but only as an undefined placeholder—damping in general is difficult to model analytically except for specific cases involving simplifying assumptions [54]. As understanding the system damping realistically was integral to the primary research objectives, estimations of the system damping were obtained empirically during the second phase of research.

The second phase of research consisted of a series of experiments that measured the behavior of several MR damper test articles both with and without applied magnetic fields. For each test unit, the damping ratios and natural frequencies were calculated from accelerometer data for both the first and second mode shapes. The variable damping ratios and stiffness properties were then built back into the original MATLAB<sup>TM</sup> model, thus giving a controllable definition of system damping and stiffness as a function of magnetic field strength.

In the third phase of research, a PID controller was created that varied the system damping properties in response to dynamic motion. The goal was to investigate how settling time was affected with a controlled magnetic field.

The remainder of this section is presented in four sub-sections that outline the research methodology in detail. Part A provides a brief overview of the initial concept of the spacecraft-damper system as it was envisioned at the start of the research effort, along with an explanation of how the system has evolved over the course of the study. Part B describes the derivation of the numerical model and presents the system's equations-of-motion. Part C describes the experiments that were used to empirically define system damping ratios and stiffness. Finally, Part D describes the simple PID controller and how the empirical data was integrated into the numerical model.

### **A. System Description**

The initial concept was a system built around the most common small spacecraft form-factor, a 3U CubeSat, 10cm x 10cm x 30cm. Unconventionally, the model was assumed to include two, oversized, 30cm x 100cm rectangular solar arrays. The solar arrays were assumed to be flexible, with the rigidity and thickness of a thin sheet of aluminum—chosen because the material properties were well understood. The damper consisted of a hollow, round cylinder 6.35mm diameter that ran the length of the solar array and was surrounded by a flexible, 22AWG conductor bonded to the tube's surface. Figure 1 and Figure 2 illustrate the initial spacecraft/damper system configuration.

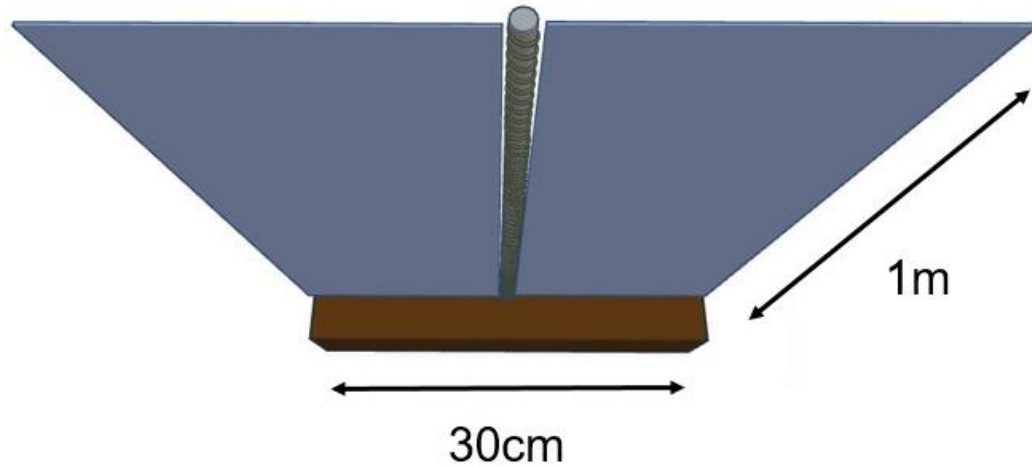


Figure 1. Right-side view of the initial satellite-damper system configuration

The cylindrical damping tube was to be filled with an undefined MR fluid at low pressure. As the tube deformed, moving with the attached flexible solar array, localized pressure changes would force relative motion between the internal fluid and the tube walls thus removing energy from the system due to friction. To make the effect controllable, the coil would carry direct current electricity applied via an upstream system controller. The associated magnetic field would thus allow for variable, intelligent damping by changing the material properties (essentially varying the damping ratio and

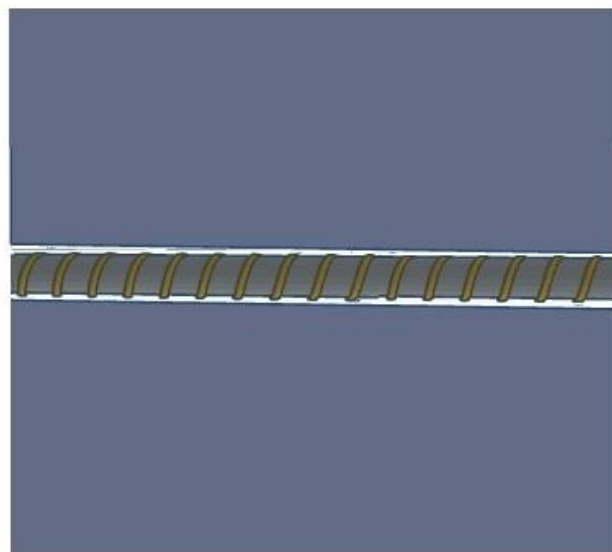


Figure 2. Top-down view of the initial satellite-damper system configuration

stiffness) of the solar array. MR fluid devices operating with flow between stationary field poles are said to be operating in a “flow” mode (as opposed to the “shear” or “squeeze flow” modes) [42].

During testing, the data showed that applying electrical current in the manner initially conceived produced thermal effects ( $I^2R$  heat losses) that were overwhelming the effects of the applied magnetic field. Additionally, the orientation of the magnetic field—axially aligned with the damper tube as opposed to perpendicularly aligned to the tube walls—may have negatively impacted the damping and stiffness changes. These effects are discussed in more detail in the Results and Conclusion sections. The system

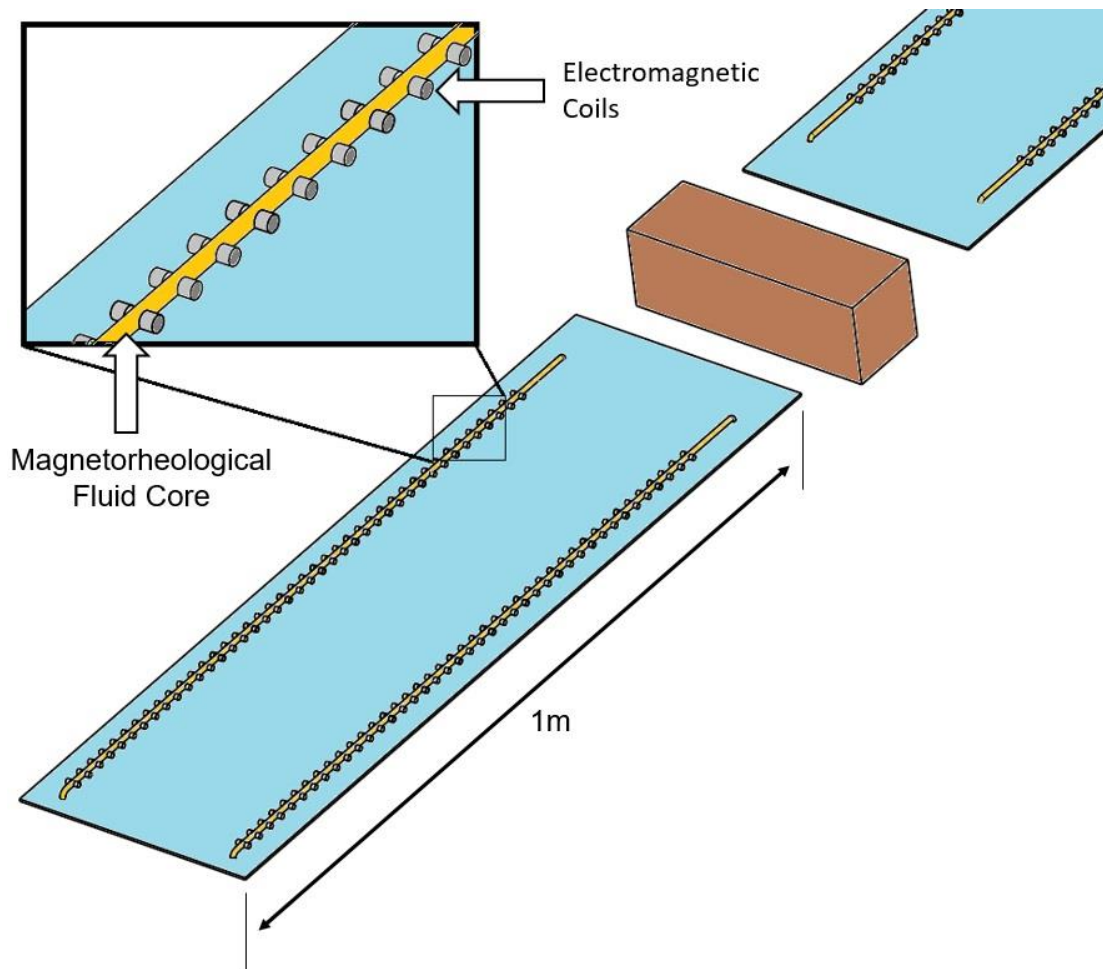


Figure 3. An illustration of one alternative spacecraft-damper system configuration

model for the remainder of testing assumed that the wrapped electrical coil surrounding the tube was replaced with discrete coils placed in proximity to the tube at regular intervals. These discrete coils, simulated by permanent magnets during most experiments, produced magnetic fields that ran perpendicular to the damper tube walls. Additionally, testing demonstrated that two tubes, one below and one above the array, with flow paths between them to allow for cyclical fluid flow, provided a stronger damping effect than with a single tube. Figure 3 provides an illustration of this alternative architecture that was adopted during testing.

One additional architecture, with a thin upper and lower MR fluid bed sandwiched between solar-cell-covered top and bottom layers with a mid-layer containing electromagnetic coils, was also considered. An illustration is provided as Figure 4. This alternative architecture may be desirable due to the increased friction associated with the greater surface area in contact with the MR fluid beds, however

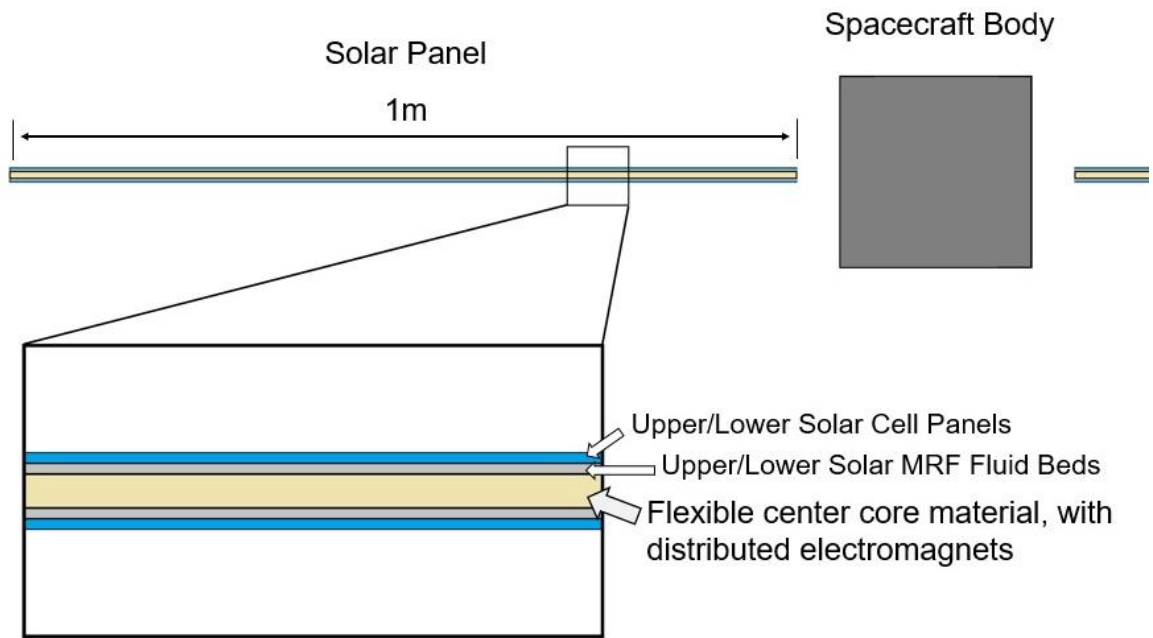


Figure 4. Profile view of a third spacecraft-damper configuration

testing this design was deemed infeasible for the current study. If constructed, this design of MRF device, with moving magnetic poles relative to the MR fluid surface, would be operating in the “shear” mode [42].



## B. Numerical Model

The equations of motion that comprise the system model are derived using a Lagrangian energy approach shown by Gorinevksy and Vukovich [55] with the exception that their application accounted for only two degrees of freedom.

To begin, a body-fixed coordinate frame (BFF) is imagined to be rigidly attached to the spacecraft body at its mass center. This arrangement is illustrated in Figure 5. The CubeSat hub of the spacecraft is modeled as a rigid body while the two solar arrays are modeled as continuous Euler-Bernoulli cantilevered beams clamped near the side of the hub but stood-off by 10cm to account for an undefined linkage.

To provide a reference for the rotational motion of the satellite, a Local Vertical, Local Horizontal (LVLH) frame is assumed to move translationally with the mass center of the spacecraft. The relationship between the LVLH frame and the BFF frame is

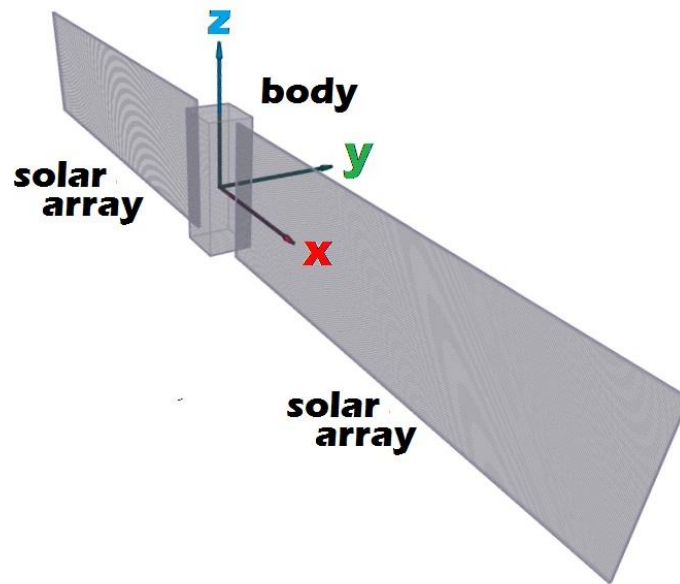


Figure 5. Body-fixed coordinate system (BFF)

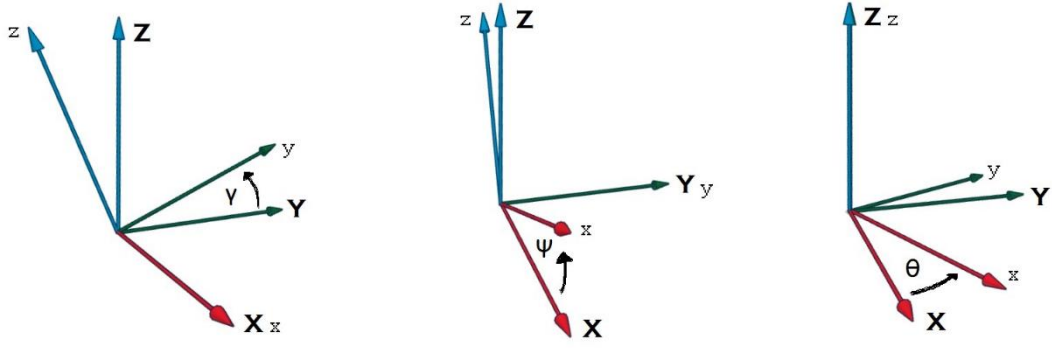


Figure 6. Relationship between the BFF and LVLH coordinate systems

illustrated in Figure 6. The  $\gamma$ ,  $\psi$ , and  $\theta$  describe respectively the roll, pitch, and yaw angles between the two coordinate systems.

The total kinetic energy of the system is described as a combination of the rotational motion of the rigid hub and the sum of both the translational and rotational motion of all points of the continuous solar arrays:

$$T = \frac{1}{2} I_{SAT_Z} \dot{\theta}^2 + \frac{1}{2} I_{SAT_Y} \dot{\psi}^2 + \frac{1}{2} I_{SYS_x} \dot{\gamma}^2 + \mu \int_a^{L+a} (\xi \cos \psi \dot{\theta} + \dot{y})^2 d\xi \quad (1)$$

where  $T$  is the kinetic energy of the system;  $I_{SAT}$  is the moment of inertia of the central spacecraft body not including the solar arrays;  $I_{SYS}$  is the moment of inertia of the combined system including the central body and both solar arrays;  $\mu$  is the linear density of the solar arrays including the MR fluid damper;  $L$  is the solar array length;  $a$  is the distance from the spacecraft mass center to the closest edge of either solar array; and  $\xi$  is a linear coordinate along the length of the solar array.

Since the spacecraft is not assumed, for the purposes of the model, to be operating within an external gravity field, the potential energy of the system results simply from the strain energy of the solar arrays according to the Euler-Bernoulli model. Some version of this potential energy function can be found in most structural mechanics textbooks as the

equation for strain energy for beam bending [56]. The version shown here is doubled to account for symmetry of the solar arrays:

$$V = EI \int_a^{L+a} \left( \frac{\partial^2 y}{\partial \xi^2} \right)^2 d\xi \quad (2)$$

where  $V$  is the potential energy of the system,  $E$  is the Young's modulus of the assumed solar array material, and  $I$  is the moment of inertia of the solar array about the neutral axis. The combined  $EI$  quantity is known as flexural rigidity [56].

In both Equations 1 and 2, the  $y$  coordinate is given as a piecewise defined function that is the summation of the product of the generalized coordinate  $q_j$ , itself a function of time, and the shape function  $\phi(\xi - a)$  for some number of assumed modes,  $n$ .

$$y(t, \xi) = \begin{cases} \sum_{j=1}^n q_j(t) \phi_j(\xi - a) & \text{for } \xi \in [a, L + a] \\ \sum_{j=1}^n -q_j(t) \phi_j(-\xi - a) & \text{for } \xi \in [-a, 0] \end{cases} \quad (3)$$

Because Equations 1 and 2 require only the positive values from  $a$  to  $L + a$  (due to symmetry), only the positive domain of Equation 3 is actually used.

The shape function  $\phi(\xi - a)$  in Equation 3 is taken to be the modal function of a uniform cantilever beam as presented by Rao [54] and is shown for clarity below as Equation 4:

$$\phi(\Delta) = \left[ \sin \beta_n \Delta - \sinh \beta_n \Delta - \frac{(\sin \beta_n L + \sinh \beta_n L)(\cos \beta_n \Delta - \cosh \beta_n \Delta)}{(\cos \beta_n L + \cosh \beta_n L)} \right] \quad (4)$$

where  $\Delta = \xi - a$ , and  $\beta_n L$  is a constant defined as shown in Table 1 for the first three assumed mode shapes.

Table 1. Values of  $\beta_n L$  for the first 3 mode shapes [54]

Assumed Mode	$\beta_n L$
n = 1	1.875104
n = 2	4.694091
n = 3	7.854757

By using the function definition shown in Equation 3 in Equations 1 and 2, and after consolidating and evaluating the resulting integrals, the kinetic and potential energy expressions are rewritten as follows:

$$T = \frac{1}{2}(I_{SAT_Z} + J_3 \cos^2 \psi) \dot{\theta}^2 + \frac{1}{2}(I_{SYS_y}) \dot{\psi}^2 + \frac{1}{2}(I_{SYS_x}) \dot{\gamma}^2 + \cos \psi \dot{\theta} \mathbf{m}^T \dot{\mathbf{q}} + \dot{\mathbf{q}}^T [M] \dot{\mathbf{q}} \quad (5)$$

$$V = \mathbf{q}^T [K] \mathbf{q} \quad (6)$$

where  $J_3 = \frac{2}{3} \mu [(L + a)^3 - a^3]$  and  $\mathbf{m}$ ,  $[M]$ , and  $[K]$  are defined as shown in Equations 7, 8, and 9.

$$\mathbf{m} = 2\mu \int_a^{L+a} \xi \phi_i(\xi - a) d\xi \quad (7)$$

$$[M] = \mu \int_a^{L+a} \phi_i(\xi - a) \phi_j(\xi - a) d\xi \quad (8)$$

$$[K] = EI \int_a^{L+a} \ddot{\phi}_i(\xi - a) \ddot{\phi}_j(\xi - a) d\xi \quad (9)$$

These expressions for the kinetic and potential energies are related through application of Lagrange's Equation to the generalized external forces,  $Q$ , acting on the system as shown:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial T}{\partial \mathbf{q}} + \frac{\partial V}{\partial \mathbf{q}} = Q \quad (10)$$

After substitution of Equations 5 and 6 into Equation 10, and after evaluating the resulting partial derivatives, the system equations-of-motion are revealed as a system of four coupled ordinary differential equations.

$$\tau_\gamma = I_{SYS_x} \ddot{\gamma} \quad (11)$$

$$\tau_\psi = I_{SYS_y} \ddot{\psi} + J_3 \cos \psi \sin \psi \dot{\theta}^2 + \sin \psi \dot{\theta} \mathbf{m}^T \dot{\mathbf{q}} \quad (12)$$

$$\tau_\theta = (I_{SAT_Z} + J_3 \cos^2 \psi) \ddot{\theta} - 2J_3 \cos \psi \sin \psi \dot{\psi} \dot{\theta} + \cos \psi \mathbf{m}^T \ddot{\mathbf{q}} - \sin \psi \dot{\psi} \mathbf{m}^T \dot{\mathbf{q}} \quad (13)$$

$$0 = 2[M] \ddot{\mathbf{q}} + 2[K] \mathbf{q} + \cos \psi \ddot{\theta} \mathbf{m} - \sin \psi \dot{\theta} \dot{\mathbf{m}} \quad (14)$$

These equations model the behavior of the system with the exception of the inherent damping—which is not accounted for (i.e. there is no  $[C]$  matrix in Equation 14. This is not entirely misleading as thin beams vibrating in a vacuum should be expected to exhibit low damping characteristics, but it is ultimately not a true representation of any real system. For the purposes of this study, which is tasked with modeling the behavior of the proposed MR fluid system, the inherent damping is assumed to be negligible relative to the contribution from the MR fluid system; any additional damping which may exist will only work to improve system performance.

To insert the effects of the MR fluid damper into the system model, a dynamic  $[C]$  matrix is artificially inserted into Equation 14. This new damping matrix is  $n \times n$ , where  $n$  represents the number of modes considered in the model, and diagonal. Its elements,  $c_{11}, c_{22}, \dots, c_{nn}$ , are functions of the applied magnetic field strength and based on the experiments described in the following sections. The revised Equation 14, after insertion of  $[C]$ , is shown below as Equation 15.

$$-[C] \dot{\mathbf{q}} = 2[M] \ddot{\mathbf{q}} + 2[K] \mathbf{q} + \cos \psi \ddot{\theta} \mathbf{m} - \sin \psi \dot{\theta} \dot{\mathbf{m}} \quad (15)$$

Equations 11-13 and 15 thus provide the complete system model with the dynamic motion of the flexible appendages approximated by the superposition of  $n$  number of mode shapes scaled by the elements of the generalized coordinate vector,  $\mathbf{q}$ . The quality of the approximation and, in turn, the usefulness of the model, is dependent upon including enough mode shapes to be useful, but not so many as to be cumbersome.

The effective modal mass,  $m_{eff_i}$ —where  $i = 1 \dots n$ —is calculated as shown in Reference [57] and provides a measure of the relative contribution of each individual mode shape to the overall dynamic motion of the system as a percentage of the total system mass,  $m_{sys}$ . In this case,  $m_{sys}$  is the combined mass of the solar array and damper. Although not calculated here, Table 2, taken from Reference [57], shows the effective modal mass of the first four mode shapes for the cantilever beams included in the system model.

Table 2. Effective modal masses for the first four mode shapes of a cantilever beam

Mode Number	Effective Modal Mass
1	0.6131 $m_{sys}$
2	0.1883 $m_{sys}$
3	0.06474 $m_{sys}$
4	0.03306 $m_{sys}$

As can be seen in Table 2, including the first 4 mode shapes would account for approximately 90% of the total system mass, the first 3 would account for 87%, and, if only the first two were included, 80% of the system mass would be accounted for.

### **C. Damping Matrix Definition**

Providing a definition for the damping matrix in the system model was complicated by two factors. First, generalized material damping properties are difficult to model mathematically due to the large number of contributing mechanisms, and because, for many of those mechanisms, the contributions are poorly understood [54]. Second, the material properties of the MRF damper described here are not static. Instead, the damping properties change in response to controlled magnetic fields. Therefore, in order to provide a definition for the damping matrix that was representative of the true system and that accurately reflected the dynamic nature of the damper, system damping was defined empirically, through a series of two controlled experiments on four test articles.

#### **1. Impact Hammer Modal Testing**

In the first experiment, three demonstration test articles were constructed and subjected to impact hammer modal testing. The three test articles were constructed from similar materials and with similar techniques, but were each targeted to evaluate slightly different operational designs.

The first test-article was constructed using a single 45cm length of clear PVC tubing. The tube had an outer diameter of 6.35mm and an inner diameter of 4.32mm. The tube was filled with 18.3g (approximately 6.1mL) of LORD MRF-132DG MR fluid and sealed with plastic caps secured in place with a small amount of general-purpose adhesive. Approximately 2.5cm lengths of heat-shrink tubing were placed over the cap-to-tube joints to provide a secondary barrier to leakage. The filled tube was then attached to a thin strip of aluminum, 63.5cm x 2cm x 0.25mm, which acted as a surrogate for the solar array material. The tube, centered on the aluminum strip, was secured in place by



Figure 7. First test article

nine small plastic cables-ties spaced at 4.8cm intervals. The cable ties were tightened to the point where they were just slightly deforming the tube at each location. A picture of the first test article is shown as Figure 7.

The second test article utilized two connected MRF filled tubes that were slightly longer than the first at 50.3cm. In this design, one tube was mounted above the aluminum strip and one below such that as the tube vibrated, one tube would always be in compression while the opposite tube was in tension. The goal was to encourage relative motion between the fluid and the tube walls. Instead of the plastic capped-ends used in the first test-unit, the tubes of the second unit were connected to sealed nylon spacers at either end, 2.54cm tall, which provided flow paths between the upper and lower tubes. A

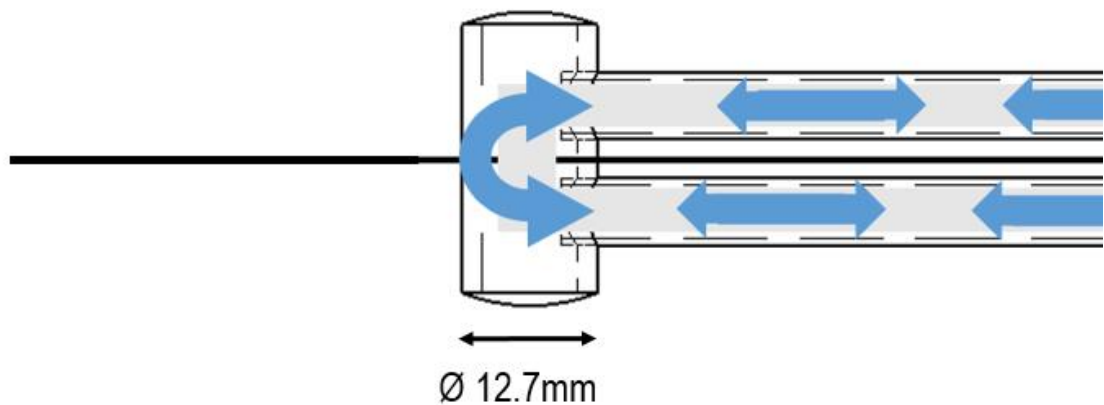


Figure 8. Internal flow paths of the second test article





Figure 9. Second test article

diagram illustrating the internal geometry of the second test article is shown as Figure 8 and a picture is included as Figure 9. This second test article was filled with 42.5g (approximately 14.2mL) of MR fluid.

The third test article was constructed using the same materials and with similar dimensions to those of the first. Again, a single 45cm-long tube was filled with MR fluid (19.7g, approximately 6.6mL), and sealed with plastic caps and shrink-tube. This third test article was then wrapped in 22AWG wire at a linear turn density of 6-7 turns/cm. The goal was to provide an integrated coil to apply a controllable magnetic field. A picture of the third test article is shown as Figure 10.

During each test, one of the three test articles was mounted on a custom-constructed test fixture; Figure 11 presents an illustration. The test fixture allowed for the mounting of the test articles under a small amount of tension (similar to deployed solar



Figure 10. Third test article

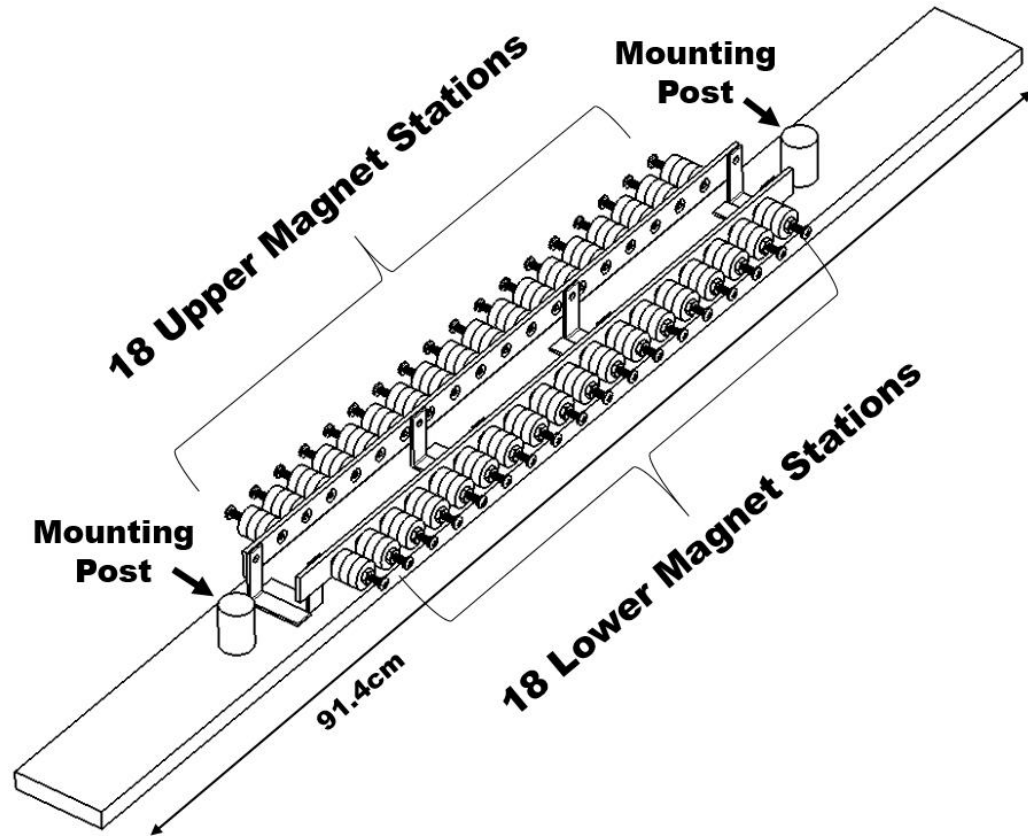


Figure 11. Custom-built test mount

arrays in space) and then clamped in a fixed-fixed configuration. Because the exact tension placed on each test article was not measured during mounting, the test articles were not removed until the completion of the entire test sequence for that test article. Once mounted, sets of external magnets were installed on the test fixture, resulting in multiple, parallel magnetic fields running the length of the MRF-filled tubes, with field lines perpendicular to the tubes' axes. The external magnets were installed on two aluminum bars, 3.2mm thick, each containing 18 stations of magnets where each station consisted of a central bolt with surrounding neodymium ring magnets. The magnets were all of the same geometry, 19.05mm diameter x 6.35mm height with a center hole 4.06mm diameter. Figure 12 is a picture of the test fixture with installed test-article and magnets—magnetic poles and field lines have been illustrated for clarity.

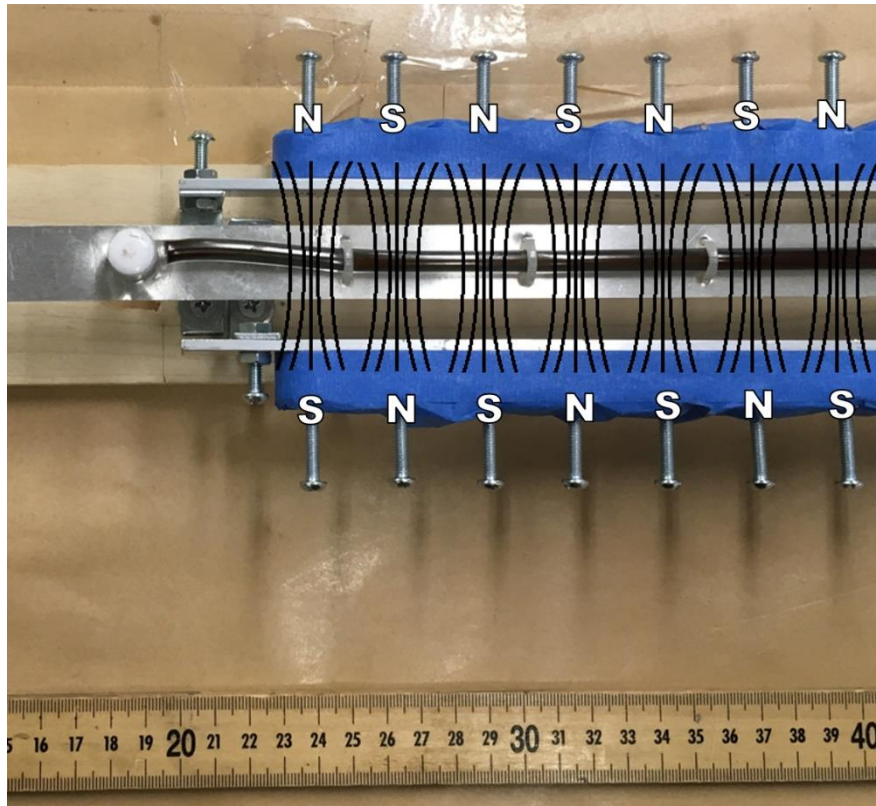


Figure 12. Second test article, in the test mount, with illustrated field lines and poles

For test articles 1 and 2, the resulting strength of the applied magnetic field was measured using a HT20 Tesla meter. The field strength was measured to be as low as 25-30mT at the centerline of the test fixture with a maximum of 200-250mT on contact with the aluminum mounting bars. Field strength on any individual magnet's surface was measured at approximately 400-500mT.

Since the third test article was built with an integrated coil, it instead relied on electrical current flow as a magnetic source as opposed to permanent magnets; it was tested with no current, 5A, and 10A. Because of the orientation of the coil relative to the tube, the magnetic field for test article 3 ran parallel to the tube axis rather than perpendicular, as was the case with test-articles 1 and 2. After modal testing was completed, to allow for testing of the magnetic field strength associated with the

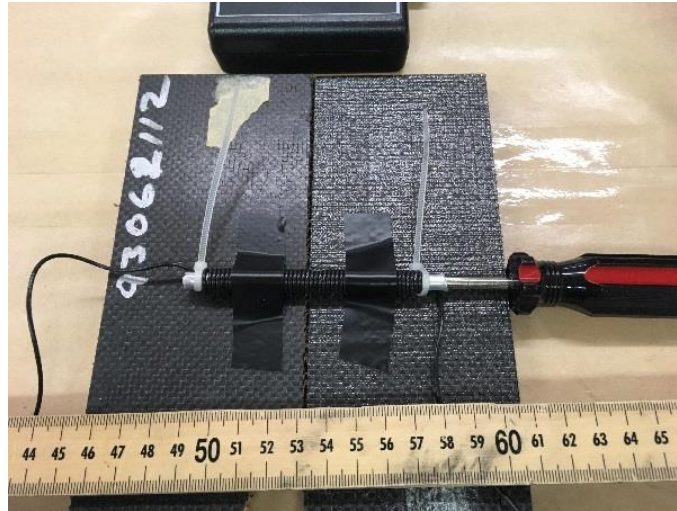


Figure 13. Demonstrator tube used to measure the third test article's field strength

integrated coil, a separate aluminum tube was wrapped in 22AWG wire at the same turn density as the third test article—a picture is shown as Figure 13—and field strength was measured with the Tesla meter. A steel screwdriver was inserted into the tube at the same time, but at the opposite end from the probe of the Tesla meter, to mimic the magnetic core the MR fluid would have provided in the actual test article. The resulting field strengths measured by the Tesla meter are shown in Table 3 at current levels of 1A, 5A, and 10A. As can be seen, the magnitude of the field produced by 10A current was approximately comparable to the field produced by the external magnets used to test the first and second test articles.

Table 3. Magnetic field produced by integrated electrical coil

Electrical Current	Field Strength
1A	3mT
5A	12mT
10A	23mT

Impact hammer modal testing was performed using a PCB Piezotronics impulse hammer, model 086C01, and three single-axis PCB Piezotronics accelerometers mounted on the test piece. The left-most and right-most mounting locations were placed

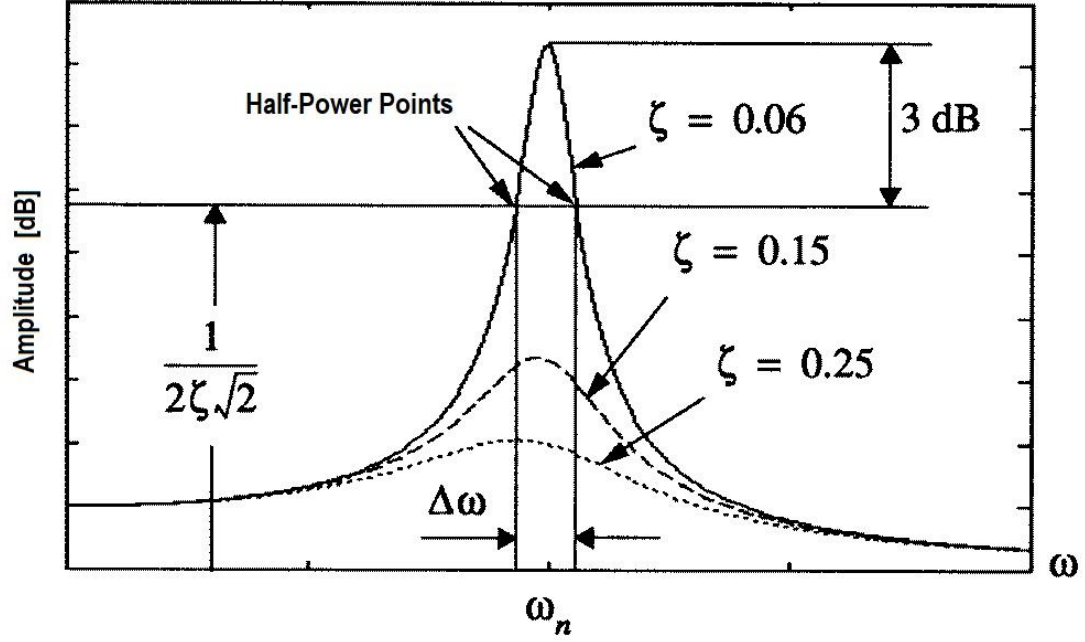


Figure 14. Half-power points of a resonant peak [64]

approximately  $L/6$  distance away from either end with an additional accelerometer mounted in the middle. Care was taken to place the second accelerometer slightly off-center to avoid the node location of the second mode shape. The test fixture was mounted such that each test article would vibrate in a direction parallel with the ground, thus minimizing the influence of gravity.

Data was recorded using LabVIEW™ to take the average of five sequential datasets for each tested condition and output phase and magnitude frequency response data for each accelerometer. Post-processing of the data was performed using MATLAB™.

From the frequency response data, damping ratios were calculated using the half-power bandwidth method as described in Rao [54] where:

$$\zeta \cong \frac{\omega_2 - \omega_1}{2 \omega_n} \quad (16)$$

In Equation 16,  $\omega_n$  represents the modal resonance peak while  $\omega_1$  and  $\omega_2$  are the frequencies associated with the “half-power points”—the points at which amplitude is  $\frac{1}{\sqrt{2}}$  peak. When amplitude is plotted in decibels, as is convention, this equates to approximately 3dB below the peak value. An illustration of the half-power points is provided as Figure 14.

Flexural rigidity (EI) was also calculated from the frequency response data using the first mode natural frequency and Equation 17 taken from Rao, [54].

$$\omega_1 = \beta_1 L^2 \sqrt{\frac{EI}{\rho AL^4}} \quad (17)$$

In Equation 17  $\beta_1 L$  is a constant value for first mode vibration of a fixed-fixed beam, taken from [54] to be 4.730041,  $\rho$  is the solar array-damper system equivalent density, and  $L$  is the system length.

## 2. Shaker Table Vibration Testing

To supplement the data recorded during the impact hammer modal testing, a second experiment was performed in which a fourth test article was constructed and mounted to an electrodynamic shaker table. Results from the hammer impact modal testing—described further in the Results section—revealed the design of the second test article, the one constructed with two MRF-filled tubes, had the most significant



Figure 15. Fourth test article, used for shaker table vibration testing



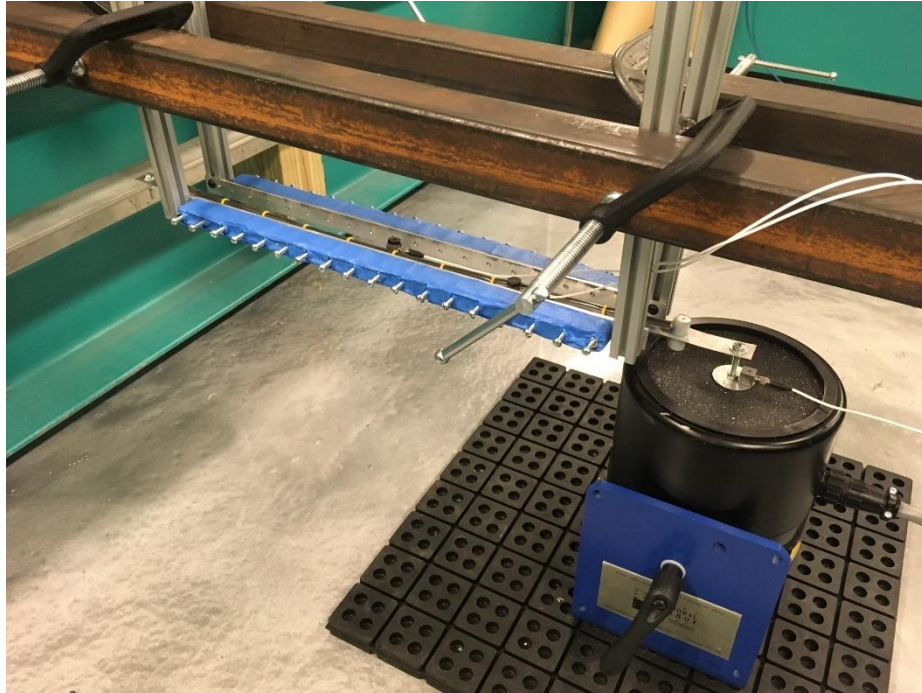


Figure 16. The fourth test article, mounted during shaker table vibration testing

improvement in damping associated with the applied magnetic field; thus the fourth test article was simply a reproduction of the second in all respects. A picture of the fourth test article used in the shaker table testing is presented as Figure 15. The fourth test article was filled with 48.4g (approximately 16.1mL) of LORD MRF-132DG.

For the experiment, the test article was mounted in a fixed-fixed beam configuration with one end bolted to the threaded mount extending from the top of the shaker. The opposite end was bolted to an aluminum beam clamped to a large, immobile, steel structure. Figure 16 shows the installed test article.

The electrodynamic shaker used during testing was produced by The Modal Shop, Inc., model 2060E. Three accelerometers were used to record data, all were made by PCB Piezotronics, model 333B30. One accelerometer recorded the excitation input; it was mounted to a washer, bolted to the top of the shaker table. Two accelerometers were mounted to the test article to capture the vibration output near the first and second anti-

node locations. Data was recorded using LabVIEW™ and then exported through a MATLAB™ interface that had been developed for use by Dr. Claudia Moreno at Embry-Riddle Aeronautical University. Post processing of the data was accomplished using MATLAB™.

Testing consisted of a sin wave frequency sweep from 5-40Hz with the goal of capturing the first three modal resonant frequency peaks. The test was first performed with an applied magnetic field—25-30mT as measured with an HT20 Tesla meter—and then repeated with the magnets removed. As the test article was being forcibly excited by the electrodynamic shaker, as opposed to the free-vibration in the impact hammer modal testing, the effects of gravity were judged to be negligible and the test article was mounted as shown in Figure 16.

After testing concluded, the input and output data was used to construct two transfer function plots, one for each of the two output accelerometers, using the MATLAB™ *tfestimate* function [58]. The frequency spectrum and transfer function plots were then used with the *fitmagfrd* function [59] to construct actual equations for the system's transfer functions that modeled as closely as possible the magnitude data resulting the *tfestimate* output. While using the *fitmagfrd* function, the user enters the desired number of poles and zeros<sup>1</sup> and the function returns the best transfer function estimate of that form matching the provided data. To evaluate the validity of the resulting transfer function estimates, the input accelerometer data was then multiplied by the transfer function estimate and plotted versus the actual output waveforms. To improve

---

<sup>1</sup> For clarity, the *fitmagfrd* function accepts arguments for the number of poles and the relative degree (the difference between the poles and zeros) of the transfer function, but it has the same effect as if entering poles and zeros.



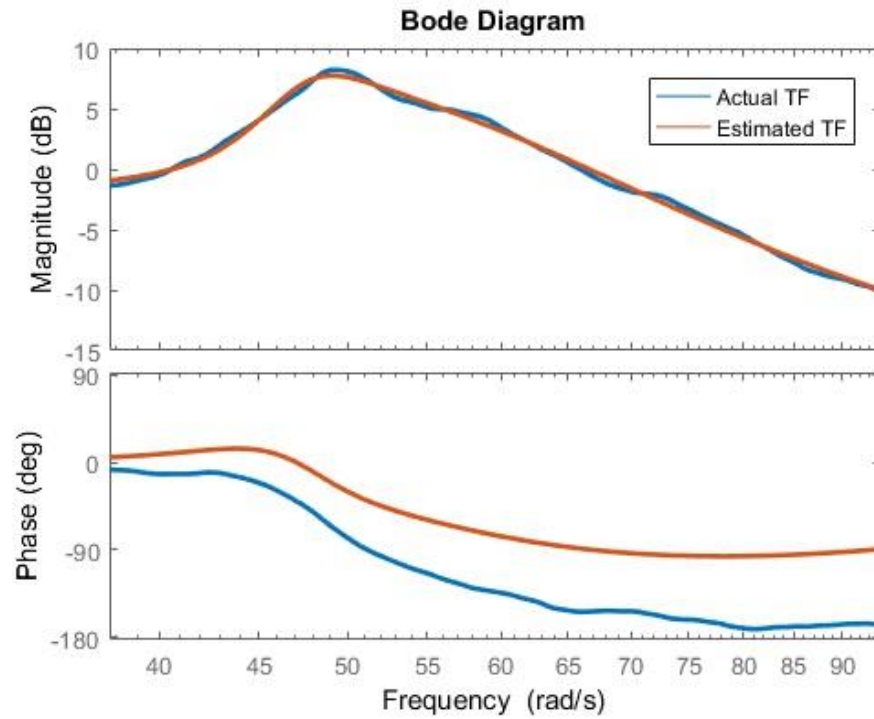


Figure 17. Bode plot—Estimated versus actual TF in the vicinity of the first mode resonant peak

the quality of the estimations in the most-relevant sections of data—the vicinity of the modal resonant peaks—the transfer functions were constructed only for those local sections as opposed to the entire 5-40Hz domain. For illustration, the estimated versus actual transfer function in the vicinity of the first mode peak for the frequency sweep performed with no magnetic field is presented as Figure 17. All transfer function plots and corresponding verification plots—the plots resulting from multiplying the input data by the estimated transfer functions—are included in Appendix A.

Once the resulting transfer functions were modeled, the MATLAB<sup>TM</sup> *damp* function [59] was used to calculate the complex conjugate pairs that represented the modal frequencies and damping ratios. These values were then compared to those calculated directly using the half-power bandwidth method as described in the previous section and stated explicitly in Equation 16.

#### D. Controller Implementation

The data from the impact hammer modal testing and the shaker table vibration experiments produced results broadly illustrating the magnitude of change in the first mode damping ratio and overall flexural rigidity associated with the application of a 25-30mT magnetic field. Since only the single magnetic field was tested, the numerical model assumes a linear relationship between the applied field and the resulting changes in damping and flexural rigidity [20].

Because testing did not reveal significant changes in the damping of the second mode shape, second mode damping is assumed to be constant. Because no usable data was gathered regarding third mode damping, the effects of the third mode vibration were discarded from the model. Additional details are provided in the Results section.

In order to establish magnetic field strength within the numerical model previously discussed, a PID controller was implemented. The controller utilizes the generalized coordinate vector  $\mathbf{q}$  and its derivative to develop an error vector, composed of first and second mode error signals as shown:

$$\mathbf{q} = \begin{Bmatrix} q_1 \\ q_2 \end{Bmatrix} \quad (18)$$

$$\mathbf{e} = \begin{Bmatrix} e_1 \\ e_2 \end{Bmatrix} \quad (19)$$

$$\dot{\mathbf{e}} = \begin{Bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{Bmatrix} \quad (20)$$

The error signals in turn drive magnetic field strength according to Equation 21:

$$FS = \left\| k_p \begin{Bmatrix} e_1 \\ e_2 \end{Bmatrix} + k_d \begin{Bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{Bmatrix} + k_I \begin{Bmatrix} \int_0^t e_1 dt \\ \int_0^t e_2 dt \end{Bmatrix} \right\| \quad (21)$$

where  $k_p$ ,  $k_d$ , and  $k_I$  are arbitrary gains, whose exact values are determined by trial-and-error. The resulting magnetic field strength ( $FS$ ) is used to drive the values of the stiffness matrix  $[K]$ , defined previously in Equation 9, and to build the damping matrix,  $[C]$ , heretofore undefined.

In building the stiffness matrix, the only changing quantity is the flexural rigidity,  $EI$ . To provide that value, Equation 22, a function of  $FS$  (in mT), is shown below. The values that form the linear equation are taken from the hammer impact modal testing.

$$EI(FS) = \left( \frac{0.028Nm}{30mT} \right) FS + 0.193Nm \quad (22)$$

As with the stiffness and mass matrices, the dynamic damping matrix,  $[C]$ , is diagonal. Its  $C_{11}$  and  $C_{22}$  elements are defined as damping coefficients that represent the associated (first mode or second mode) damping-ratio-percentage of critical damping, where critical damping is defined by Equation 23 from Rao, [54].

$$C_c = 2\sqrt{km} \quad (23)$$

In defining  $C_{11}$  and  $C_{22}$ ,  $k$  and  $m$  are the associated elements of the modal stiffness and modal mass matrices which are also diagonal.

The damping ratios multiplied with the modal critical damping values are defined by the testing results. For the second mode,  $\zeta_2$  is a constant 0.098—taken as the mean average of the two MATLAB<sup>TM</sup> estimations and the two values calculated using the half-power bandwidth method. For the first mode,  $\zeta_1$  is dynamic, a function of field strength as shown in Equation 24. The values that define the linear equation are taken as the mean average of the limits from the hammer impact modal tests and the shaker table vibration testing.

$$\zeta_1(FS) = \frac{0.033}{30mT} FS + 0.060 \quad (24)$$

## IV. RESULTS

### A. Impact Hammer Modal Testing

Data from the impact hammer modal testing of the first and second test articles, the two that were constructed without integrated coils, show an increase in flexural rigidity,  $EI$ , and first-mode damping when a 25-30mT external magnetic fields was applied. The data are inconclusive regarding damping in the second and third modes. The third test article, the one constructed with an integrated, electric coil, demonstrated decreased rigidity when current was applied; this conflicted with the data from the first two, however the test article did demonstrate increased damping when current (i.e. a magnetic field) was applied.

Initial testing of the first unit, consisting of a single MRF-filled tube, produced the frequency response function (FRF) shown in Figure 18. The FRF shown was recorded

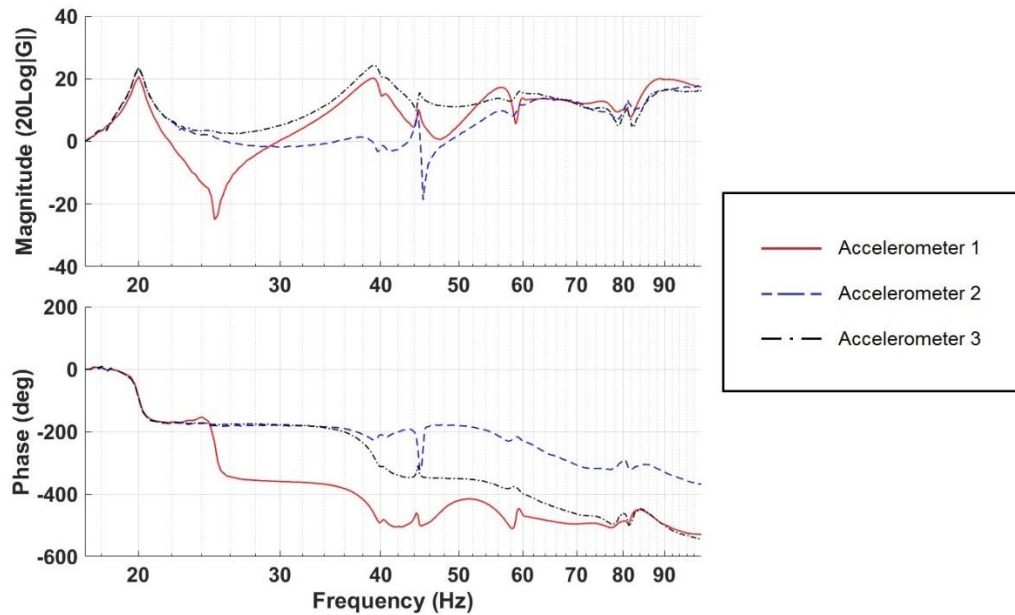


Figure 18. FRF resulting from impact hammer modal testing of the first test article

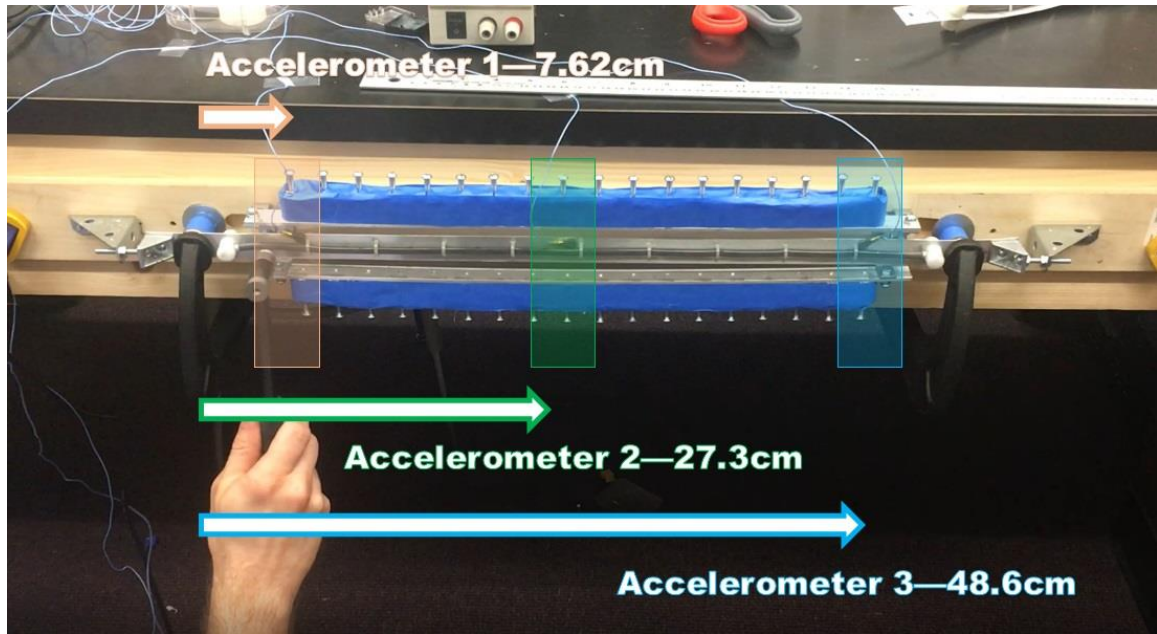


Figure 19. Accelerometer mounting locations, overall length was 55.9cm

with no magnetic field, and three accelerometers mounted at the locations marked in Figure 19. The overall length from fixed-post to fixed-post was 55.9cm.

In Figure 18, a first mode resonant peak is clearly visible at approximately 20Hz. A similar peak is noted at a second mode resonant frequency of approximately 39Hz, this is denoted by the strong response on the first and third accelerometers and a weak response on the second which was located near the anti-node (center) for that mode shape. The third mode resonant frequency peak (and those of the higher modes) are not as clearly identifiable as those of the first two although candidate peaks may be seen near 60, 80, and 90Hz.

A 25-30mT magnetic field was then applied to the test article, and the experiment repeated. Figure 20 shows both the original FRF and the shifted function following the application of the magnetic field. The wide frequency span shown in the figure makes it difficult to see the details of the changes to the higher mode shapes, but even with the

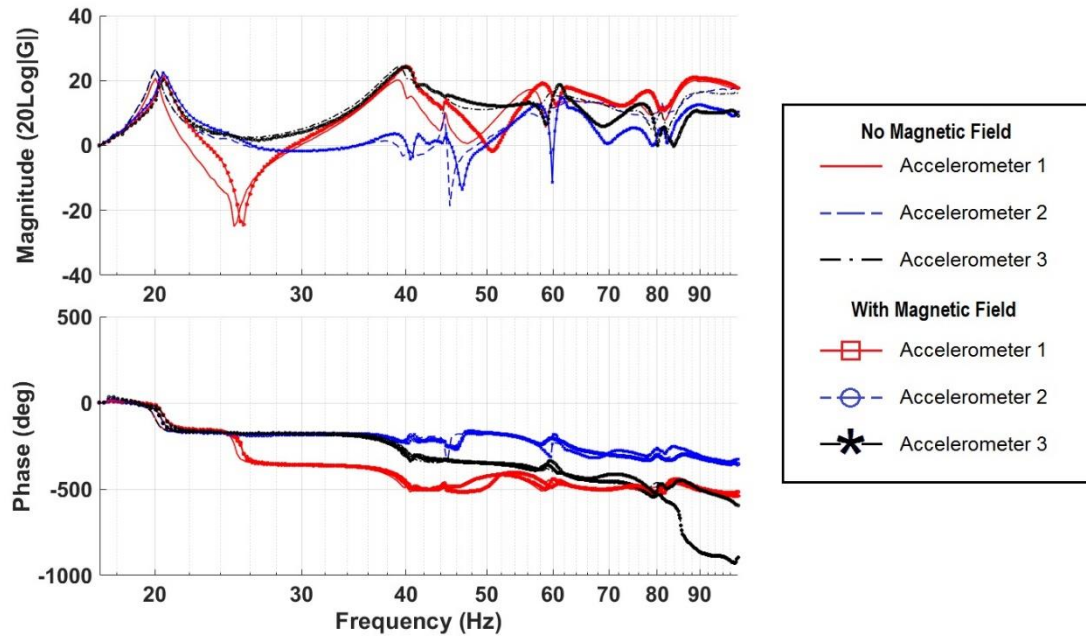


Figure 20. FRF of first test article with external 25-30mT field

dense information presented, the first mode resonant peak can be seen to be slightly attenuated and shifted to the right, towards the high frequency domain. For clarity, the first mode resonant peak, both with and without the magnetic field is shown as an isolated Bode plot in Figure 21.

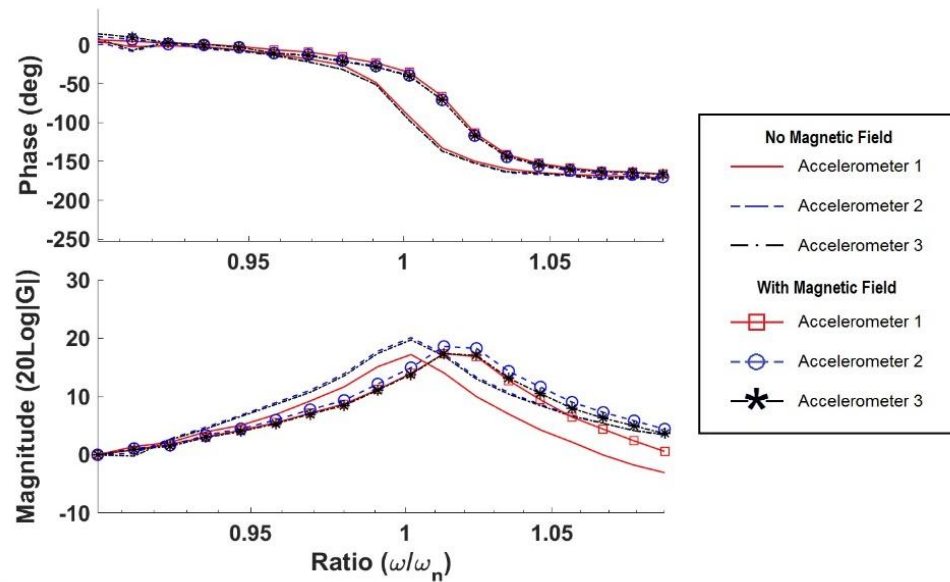


Figure 21. Bode plot—First test article, first mode resonant peak shift

Table 4. Impact hammer modal testing—First test article changes in system properties

	No Magnetic Field	25-30mT Field
Flexural Rigidity ( $EI$ )	$0.107Nm^2$	$0.112Nm^2$
Damping Ratio	0.012	0.013

Figure 21 shows more clearly the rightward shift of the first mode resonant frequency from approximately 20Hz to 20.5Hz—a shift indicating a slight increase in overall rigidity (stiffness). A slight reduction in the slope of the phase-angle shift can also be seen, indicating an increase in first mode damping ratio. Values for the changes in flexural rigidity and damping ratio were calculated using Equations 18 and 17 as discussed in the Methods section and are presented in Table 4.

Unfortunately, the distorted shape of the second mode resonant peak and the inability to positively identify the third mode peak made it impossible to establish values for the higher mode damping ratios.

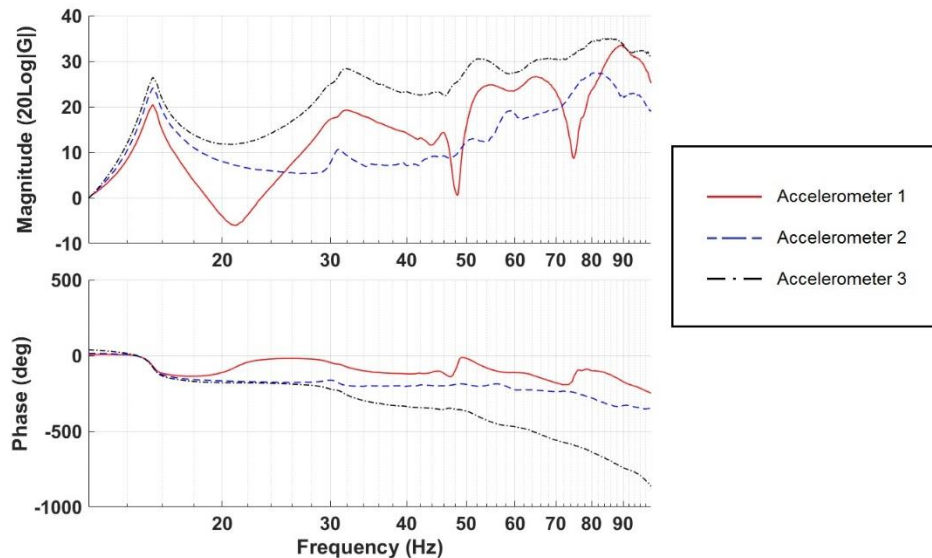


Figure 22. FRF resulting from impact hammer modal testing of the second test article

The second test article, constructed with upper and lower MRF-filled tubes connected at both ends to allow for improved flow relative to a single tube, demonstrated similar behavior to the first test article in terms of rigidity. The damping response was noticeably more pronounced in the first mode relative to the first test article. As with the first test article, distortion of the second mode shape prevented the direct calculation of the second damping ratio, and third mode peak was not identifiable. The FRF for the second test article is shown as Figure 22. As with the first article, the FRF shows a recognizable peak at the first mode resonant frequency; in this second case, it's slightly lower than the first at 15.4Hz. A Bode plot of the first mode resonant peak is shown as Figure 23 along with the resulting shift when the magnetic field was applied.

With the applied magnetic field, the frequency shifted to about 16.5Hz. Also visible in Figure 23 is a significant shallowing of the slope of the phase-angle. This shallowing is more significant than that seen with the first test article and indicates a greater increase in first mode damping ratio.

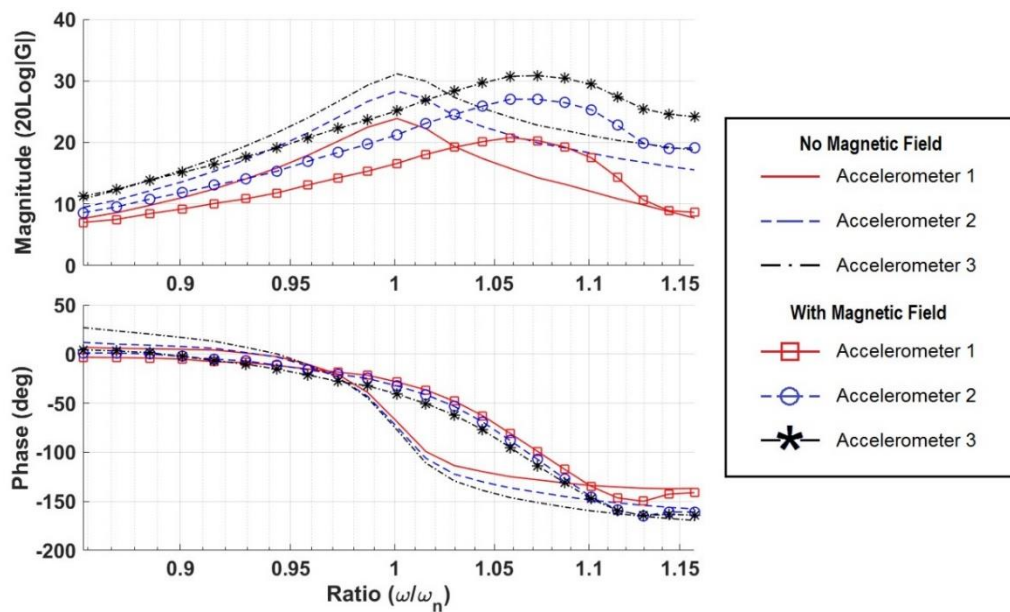


Figure 23. Bode plot—Second test article, first mode resonant peak



Table 5. Impact hammer modal testing—Second test article changes in system properties

	No Magnetic Field	25-30mT Field
Flexural Rigidity (EI)	$0.193Nm^2$	$0.221Nm^2$
Damping Ratio	0.022	0.040

The changes in flexural rigidity and first mode damping ratio observed in the second test article were calculated in the same manner as with the first test article. The results are presented in Table 5. The magnitude of change, both in flexural rigidity and damping, was more significant with the second test article than the first.

The third test article, constructed with an integrated electrical coil, was tested at current levels of 0, 5A, and 10A. The initial FRF associated with the third test article is shown as Figure 24. As with the first two test articles, the first mode resonant peak is clearly visible, in this case at 14.5Hz.

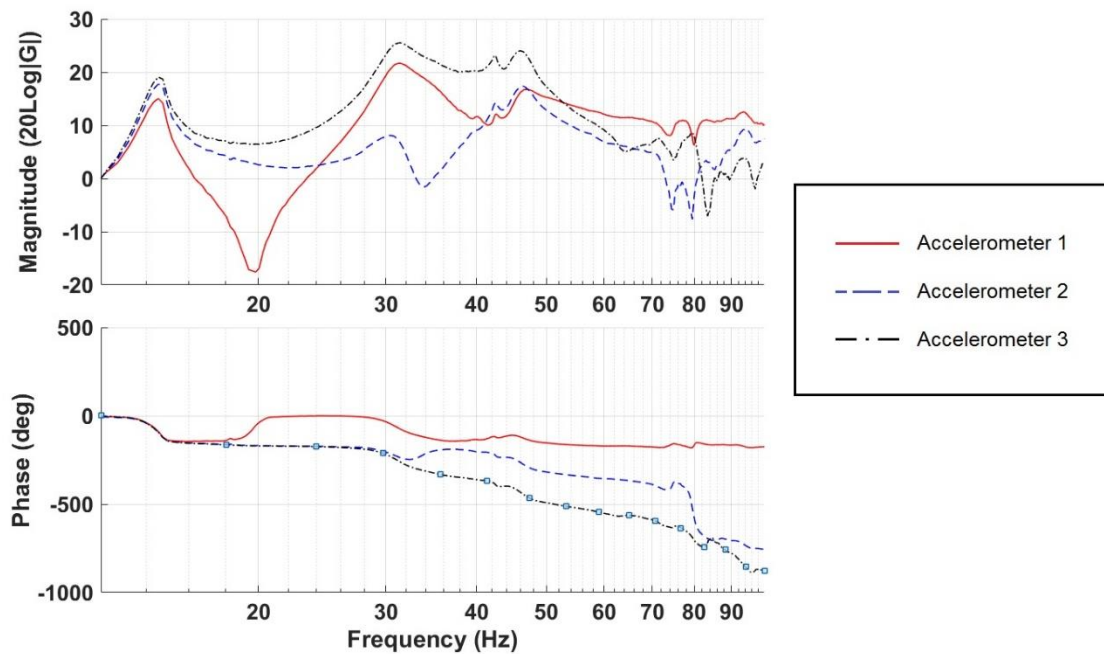


Figure 24. FRF resulting from impact hammer modal testing of the third test article

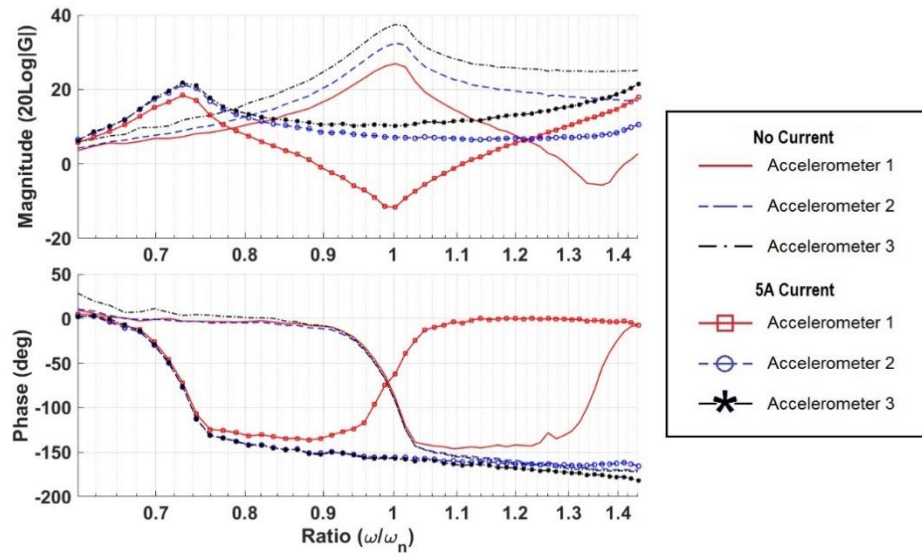


Figure 25. Bode plot—Third test article, first mode resonant peak shift, 0 to 5A

When electrical current was applied, initially 5A, the first resonant peak shifted to a lower frequency by approximately 4Hz to 10.5Hz. This was both a more significant shift than seen with either of the first two test articles and in the opposite direction. The shift is illustrated in the Bode plot shown as Figure 25. The application of further electrical current, 10A, did not significantly shift the location of the mode peak. The shift from 5A to 10A is illustrated in Figure 26.

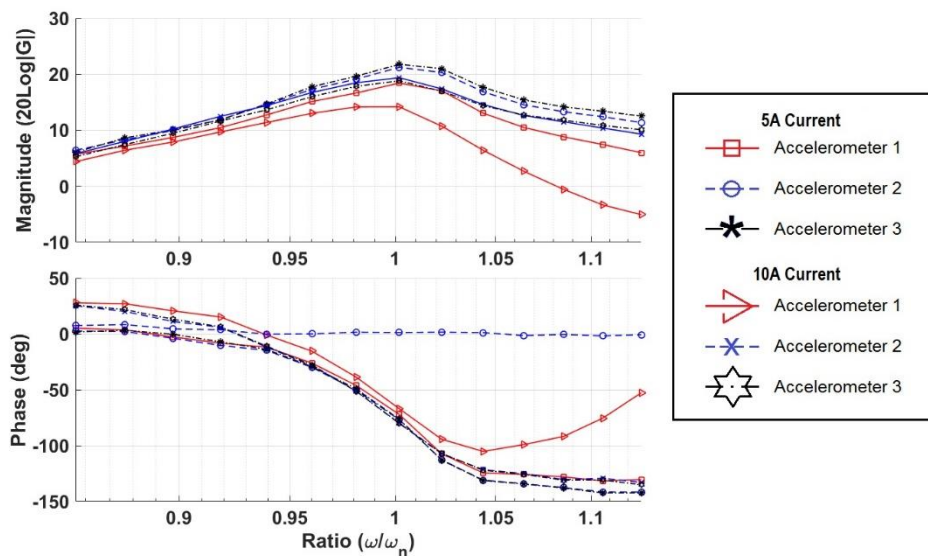


Figure 26. Bode plot—Third test article, first mode resonant peak shift, 5A to 10A

## **B. Shaker Table Vibration Testing**

Shaker table vibration testing of the fourth test article produced results broadly similar to those of the impact hammer modal testing. Notably, the modal natural frequencies that could be easily identified were lower (by approximately one-half) than those of the hammer impact modal testing. This was most likely due to the difference in how the test articles were mounted as described in the Methods section which resulted in additional tension on the hammer impact modal test articles, and a slightly longer length between fixed ends with the shaker table vibration test article—61.6cm versus 55.9cm. As with the hammer impact modal testing, both the stiffness and first mode damping increased when subjected to an applied magnetic field. Damping of the second mode shape did not significantly increase. Because the second test article, the one constructed with two MRF-filled tubes and a flow path between them, showed the strongest response during hammer impact modal testing, the fourth article was constructed mimicking that design in all respects.

Figure 27 shows the two transfer functions (two output accelerometer locations) resulting from a sin wave frequency sweep of 5-40Hz. As can be seen, the first mode resonant peak is immediately visible, and the second mode resonant peak is denoted by a strong response on accelerometer 1, mounted at the 2<sup>nd</sup> mode anti-node location—about

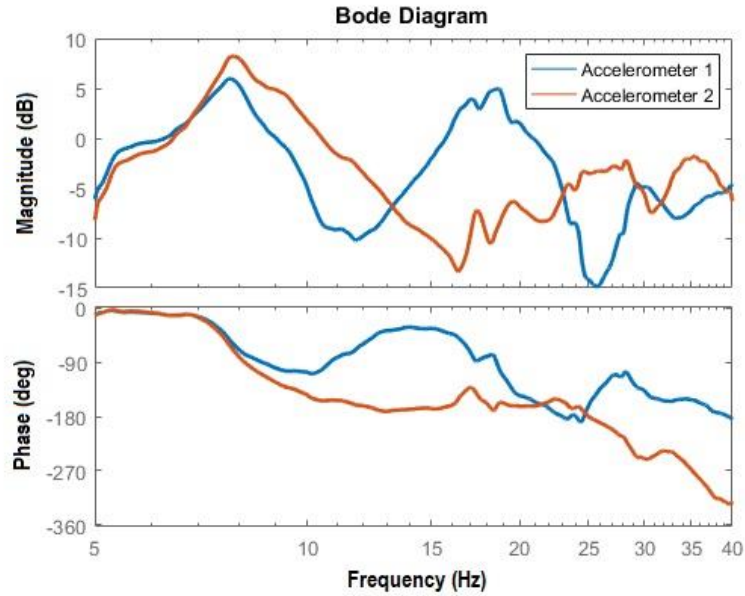


Figure 27. Frequency sweep of 5-40Hz with the fourth test article; no magnetic field

17.9cm from the edge of the 61.6cm distance between the fixed posts as shown in Figure 28—and a weak response on accelerometer 2, mounted at the node—approximately the test article center. The third mode resonant peak could not be clearly identified, likely due to its small amplitude.

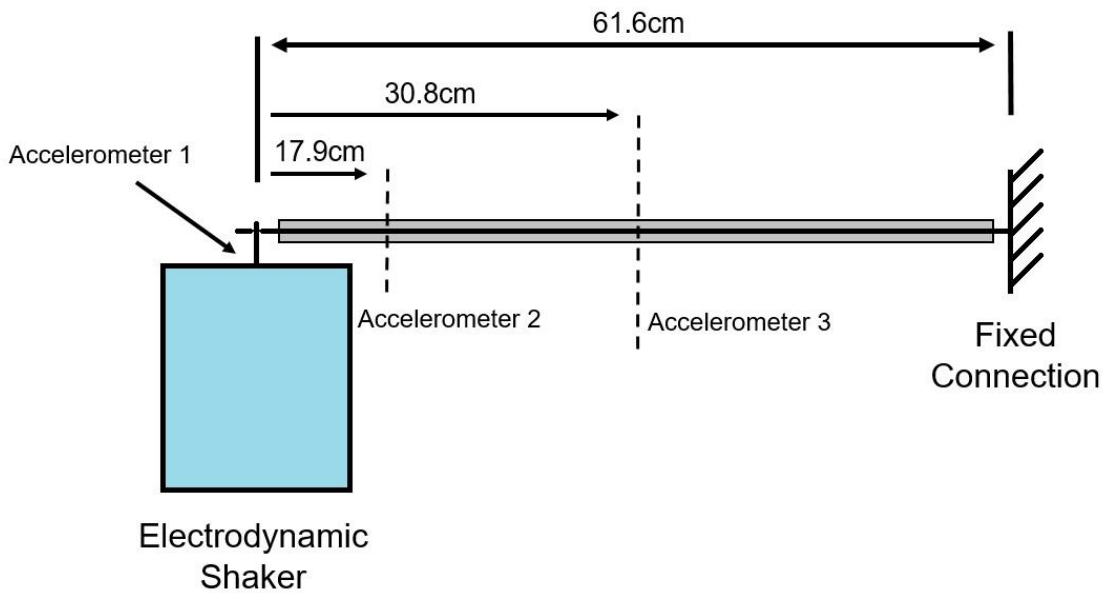


Figure 28. Experimental setup for the shaker table vibration testing

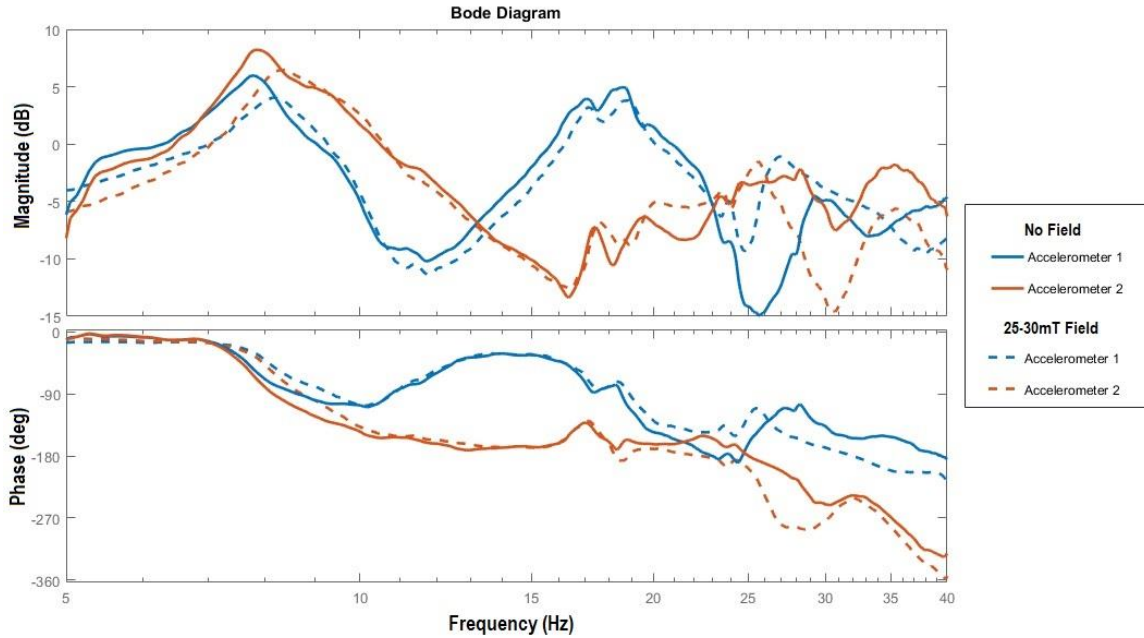


Figure 29. Frequency sweep of 5-40Hz with the fourth test article; 25-30mT applied magnetic field

When the experiment was repeated with a 25-30mT magnetic field applied to the test article, the frequency response shifted to the right, indicating an increase in overall stiffness of the test article. The first mode resonant peak shifted from 7.80Hz to 8.30Hz, 6.4% higher. The second mode peak shifted only slightly, from about 18.67Hz to 18.88Hz, about 1.1% higher. The Bode plot illustrating the shifted curve with the magnetic field applied is shown as Figure 29.

Damping for the first and second mode shapes, both with and without magnetic fields, was estimated by approximating the transfer functions of the two output accelerometers (mounted on the test article) relative to the input accelerometer mounted to the top of the shaker table, and then calculating the natural frequencies and damping ratios associated with each pole of the transfer function.<sup>2</sup> The poles with natural

<sup>2</sup> A more detailed explanation of how MATLAB™ was used to estimate the transfer functions and a discussion of the validity of those estimations is provided in the Methodology section.

Table 6. First and second mode damping ratios from shaker testing

	$\zeta_1$ Estimated by MATLAB™	$\zeta_1$ Half-Power Bandwidth	$\zeta_2$ Estimated by MATLAB™	$\zeta_2$ Half-Power Bandwidth
No Magnetic Field	0.070	0.089	0.11	0.084
25-30mT Field	0.11	0.13	0.11	0.087

frequencies located in the vicinity of the resonant mode peaks illustrated in the transfer function plots were taken to be the poles associated with those mode shapes. Those estimated values were then compared to values calculated using the half-power bandwidth method (Equation 16) based on the shape of the waveforms in the transfer function plots. The damping ratios for the first and second modes are presented in Table 6.

### C. Completed System Description and Performance

Simulation results with the empirically derived damping properties and the PID controller described in the Methods section, demonstrate reduced settling times and reduced peak amplitudes in both the first and second modes. The degree of improvement is dependent on the maximum strength of the assumed magnetic field and the initial modal amplitudes. The results of multiple simulation runs show that the PID controller allows for similar performance to a system that is constantly energized during maneuvers, but at a fraction of the required power. Figure 30 illustrates this system behavior—in terms of field strength and modal amplitudes of the first and second mode—during a two-axis slewing maneuver; plots are shown for the no field, PID controlled field, and constant maximum field cases. The fields shown are a maximum of 50mT.

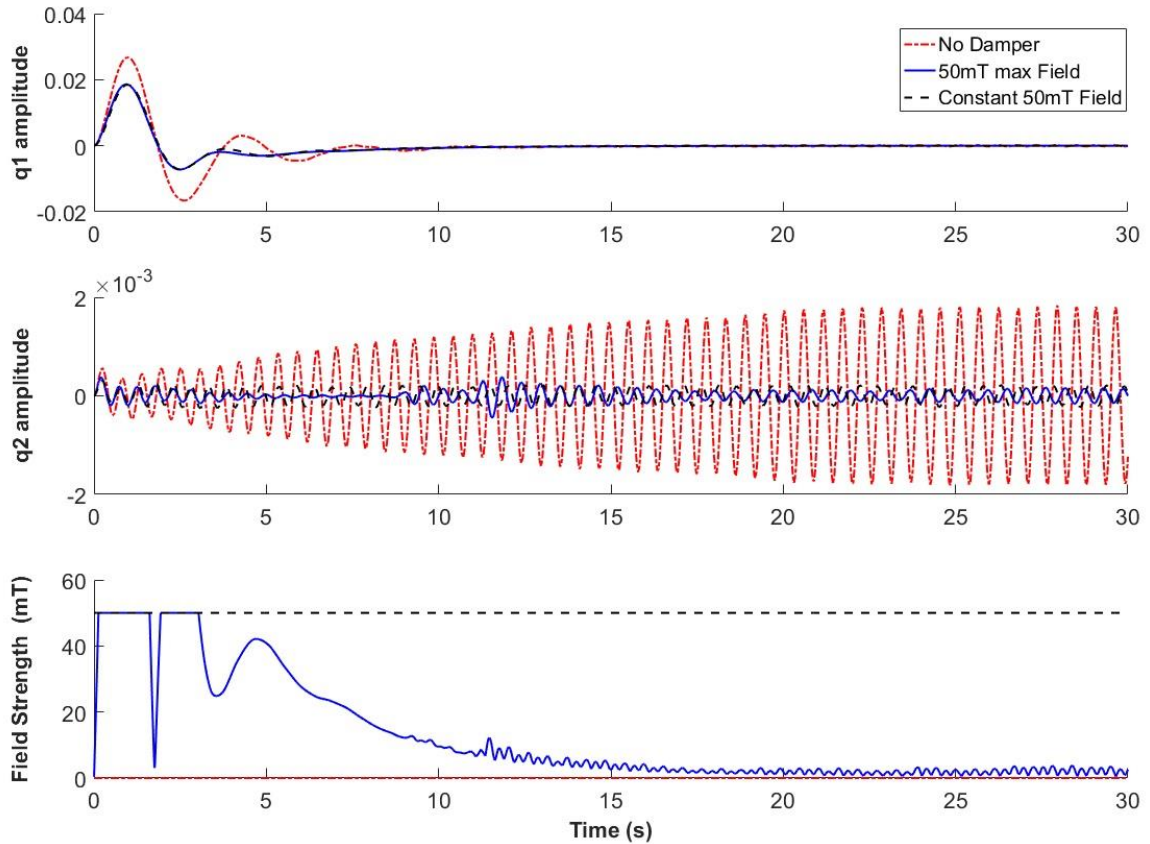


Figure 30. Simulation results from a two-axis slewing maneuver

The 30-second simulation shown in Figure 30 represents the spacecraft model maneuvering from an initial orientation of  $\gamma = \frac{\pi}{2} \text{ rad}$ ,  $\psi = \pi \text{ rad}$ ,  $\theta = 2\pi \text{ rad}$ , to a final orientation of  $\gamma = 0 \text{ rad}$ ,  $\psi = \pi \text{ rad}$ ,  $\theta = \pi \text{ rad}$ . As can be seen, the first mode amplitude,  $q_1$  satisfies a 2% settling time criteria, meaning 2% of peak  $q_1$  amplitude, within 10.66 seconds with the damper versus 12.48 seconds without the damper, a 14% improvement; the peak amplitude is also attenuated by 30%. Second mode vibration continues, at a low amplitude, beyond the duration of the simulation, but the magnitude has been attenuated by approximately 90%.

The same simulation was also run with a de-energized damper—in effect, one with a 0mT field—and a damper with a PID controlled 100mT applied magnetic field.

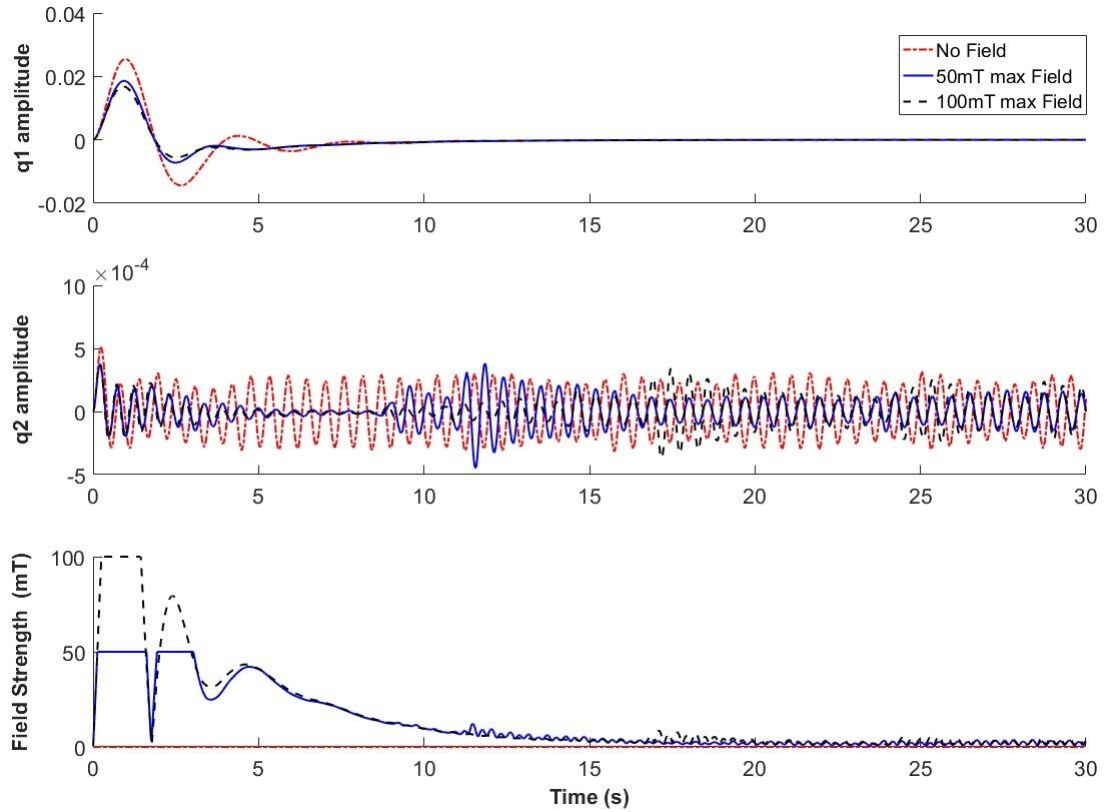


Figure 31. Two-axis slewing maneuver with three different field strengths



The goal was to understand the impact of magnetic field strength on the operation of the damper. The results are presented as Figure 31.

The results in Figure 31 illustrate close agreement between the dampers magnetized with the 100mT field and the 50mT field. Both dampers show approximately the same settling time versus the un-magnetized damper, however the 50mT damper has a 27% reduced peak amplitude in the first mode and the 100mT damper has a 34% reduced peak amplitude. All dampers have similar amplitude peak residual vibrations in the second mode that continue beyond the simulation, but the dampers with the applied magnetic fields are slightly lower on average. The complete results for both demonstration simulation runs are given in Table 7.

Table 7. Simulation results during a two-axis slewing maneuver

	1 <sup>st</sup> Mode Peak Amplitude	1 <sup>st</sup> Mode Settling Time	2 <sup>nd</sup> Mode Peak Amplitude	2 <sup>nd</sup> Mode Mean Amplitude
No Damper	0.0267	12.48s	0.0018	0.00088
Un-magnetized Damper	0.0255	10.33s	0.00051	0.00017
50mT PID	0.0186	10.66s	0.00038	0.00010
50mT Constant	0.0184	10.42s	0.00031	0.00013
100mT PID	0.0168	10.72s	0.00037	0.00010

## **V. CONCLUSIONS AND FUTURE WORK**

This thesis investigated how effective an MRF damper would be when used in the application of a small spacecraft with flexible appendages. To answer that question, the spacecraft system was modeled numerically, with the damping properties of the model defined by two experiments performed on prototypical MRF test articles. The results of simulations run using the empirically-informed model, indicate that the MRF system described in this thesis would improve platform stability. However, significant engineering challenges remain to be overcome.

More specifically, the results of this study indicate that MRF technology of the type demonstrated here could be used to provide controllable damping—the model simulations demonstrate clear reductions in modal amplitudes and settling times. Reasonably, it can be inferred that if the system were expanded to include multiple dampers working in tandem, a more controllable damping response could be provided that would optimize for desired effects or work to damp more complicated dynamic motion than that presented here—torsional vibration being one example. However, the results of this study further indicate that the configuration presented at the start of this thesis is not the optimal path forward for the implementation of these possibilities.

The impact hammer modal test data associated with the third test article, the one built with an integrated electric coil, illustrate a significant potential issue with the 6.35mm tube-style design that was tested—that issue being the power requirements and waste heat production. For the tested configurations, in order to produce strong magnetic fields, current flows were significant at 5A and 10A. However, when tested at those current levels, stiffness decreased. These data contradicted the results found when using

the permanent magnets where the stiffness was found to be increasing. Part of the cause of this disagreement may be due to the difference in magnetic field orientation—axial alignment as opposed to perpendicular alignment (a question worthy of more detailed study in future investigations)—however a likely contributor was the waste heat ( $I^2R$  losses) associated with the electrical coil. The coil was noticeably warm post-testing. The data likely indicate that, at least in part, temperature effects were dominating the magnetic field effects. Future investigations may benefit from designs that focus on minimizing tube diameter and thus reducing the required field strengths or investigating alternative MRF architectures. The fluid-bed design described briefly in the System Description section, operating in the shear mode as opposed to the flow mode of this study, may provide a starting point for those efforts.

Additionally, more traditional MRF piston-style isolators, similar to automobile MRF suspensions, may be able to provide comparable damping at lower power levels. Reference [21] for example, presents devices using less than 1A and 5W. If they could be miniaturized for the small spacecraft application they may provide effective damping at power levels more approachable for the small spacecraft model which often have total power budgets below 10W [33, 35]. Whatever path forward MRF damping is to take in the small satellite market, it is paramount that the systems be as efficient as possible.

Beyond new designs, future research would also benefit from further investigation of the following three areas that were only discussed in passing in this thesis: 1) further definition of the relationship between field strength and damping—the linear relationship assumed in this study is an acknowledged simplification; 2) optimization of materials and MR fluids for space flight applications—MRF-132DG was chosen for availability, but

due to temperature limitations is not optimal for spaceflight; and 3) alternative controller architectures.

## REFERENCES

- [1] SpaceWorks Enterprises, "Nano/Microsatellite Launch Demand Assessment 2011," 2011.
- [2] SpaceWorks Enterprises, "Nano/Microsatellite Market Forecast, 8th Edition," 2018.
- [3] S. Asmar and S. Matousek, "Mars Cube One (MarCO): Shifting the Paradigm in Relay Deep Space Operations," in *SpaceOps 2016 Conference*, Daejeon, Korea, 2016.
- [4] NASA, "Lunar Flashlight," [Online]. Available: [https://www.jpl.nasa.gov/cubesat/missions/lunar\\_flashlight.php](https://www.jpl.nasa.gov/cubesat/missions/lunar_flashlight.php).
- [5] L. McNutt, L. Johnson and D. Clardy, "Near-Earth Asteroid Scout," in *AIAA SPACE 2014 Conference and Exposition*, San Diego, CA, 2014.
- [6] NASA, "NASA CubeSat Launch Initiative," 2017.
- [7] H. Heidt, J. Puig-Suari, A. S. Moore, S. Nakasuka and R. J. Twiggs, "A New Generation of Picosatellite for Education and Industry Low-Cost Space Experimentation," in *14th AIAA/USU Conference on Small Satellites*, Logan, UT, 2000.
- [8] M. Swartwout, "CubeSats and Mission Success: 2017 Update," in *2017 Electronics Technology Workshop*, NASA Goddard Space Flight Center, 2017.
- [9] R. B. Noll, J. J. Deyst and C. H. Spenny, "A Survey of Structural Flexibility Effects on Spacecraft," in *AIAA 7th Aerospace Sciences Meeting*, New York, NY, 1969.
- [10] T. P. Sales, D. A. Rade and L. C. G. de Souza, "Passive Vibration Control of Flexible Spacecraft Using Shunted Piezoelectric Transducers," *Aerospace Science and Technology*, vol. 29, no. 1, pp. 403-412, August 2013.
- [11] M. M. Elmadany, K. A. Alsaif, M. A. Foda and A. A. Albedah, "Active Vibration Control of Satellite Panels using Piezoelectric Actuators and Sensors," *Recent Advances in Automatic Control, Modelling, and Simulation*, pp. 13-19, 2013.
- [12] L. Shaobo, Z. Jinlong, S. Junwei, J. Zongxiang and H. Yaobin, "Design and Experiment of Magnetorheological Rotary Damper for Solar Array Vibration Control," in *28th Chinese Control and Decision Conference*, Yinchuan, China, 2016.
- [13] P. W. Likins, "Dynamics and Control of Flexible Space Vehicles," National Aeronautics and Space Administration, Jet Propulsion Laboratory, Pasadena, CA, 1969.
- [14] G. Song and B. N. Agrawal, "Vibration Suppression of Flexible Spacecraft During Attitude Control," *Acta Astronautica*, vol. 49, no. 2, pp. 73-83, 2001.
- [15] H. Bang, C.-K. Ha and J. H. Kim, "Flexible spacecraft attitude maneuvering by application of sliding mode control," *Acta Astronautica*, vol. 57, no. 2005, pp. 841-850, 2005.
- [16] R. Zhang, T. Li and L. Guo, "H-Infinity Control for Flexible Spacecraft with Time-Varying Input Delay," *Mathematical Problems in Engineering*, vol. 2013, p. Article ID 839108, 2013.

- [17] J. Zheng, S. P. Banks and H. Alleyne, "Optimal attitude control for three-axis stabilized flexible spacecraft," *Acta Astronautica*, vol. 56, no. 2005, pp. 519-528, 2005.
- [18] M. Azadi, S. Fazelzadeh, M. Eghtesad and E. Azadi, "Vibration suppression and adaptive-robust control of a smart flexible satellite with three axes maneuvering," *Acta Astronautica*, vol. 69, no. 2011, pp. 307-322, 2011.
- [19] A. Spaggiari, "Properties and applications of Magnetorheological fluids," *Frattura ed Integrità Strutturale (Fracture and Structural Integrity)*, vol. 23, no. 2013, pp. 57-61, 2013.
- [20] M. Kciuk and R. Turczyn, "Properties and application of magnetorheological fluids," *Journal of Achievements in Materials and Manufacturing Engineering*, vol. 18, no. 1-2, pp. 127-130, 2006.
- [21] J. Wang and C. Meng, "Magnetorheological Fluid Devices: Principles, Characteristics, and Applications in Mechanical Engineering," *Proceedings of the Institution of Mechanical Engineers, Part L: Journal of Materials: Design and Applications*, vol. 215, no. 3, pp. 165-174, 2001.
- [22] H. Helvajian and S. W. Janson, Eds., *Small Satellites: Past, Present, and Future*, El Segundo, California: The Aerospace Press, 2008, p. 47.
- [23] M. N. Sweeting, "Modern Small Satellites--Changing the Economics of Space," *Proceedings of the IEEE*, vol. 106, no. 3, pp. 343-361, 2018.
- [24] M. Swartwout, "CubeSat Database," 2018. [Online]. Available: <https://sites.google.com/a/slu.edu/swartwout/>.
- [25] E. Gill, P. Sundaramoorthy, J. Bouwmeester, B. Zandbergen and R. Reinhard, "Formation flying within a constellation of nano-satellites: The QB50 mission," *Acta Astronautica*, vol. 82, no. 2013, pp. 110-117, 2013.
- [26] M. Urban, O. Nentvich, V. Stehlikova, T. Baca, D. Vladimir and R. Hudec, "VZLUSAT-1: Nanosatellite with miniature lobster eye X-ray telescope and qualification of the radiation shielding composite for space application," *Acta Astronautica*, vol. 140, no. 2017, pp. 96-104, 2017.
- [27] J. Virgili and P. C. Roberts, "Delta-DSat, a QB50 CubeSat mission to study rarefied-gas drag modelling," *Acta Astronautica*, vol. 89, no. 2013, pp. 130-138, 2013.
- [28] L. Johnson, M. Whorton, A. Heaton, R. Pinson, G. Laue and C. Adams, "NanoSail-D: A Solar Sail Demonstration Mission," *Acta Astronautica*, vol. 68, no. 2011, pp. 571-757, 2010.
- [29] V. Lappas, N. Adeli, L. Visagie, J. Fernandez, T. Theodorou, W. Steyn and M. Perren, "CubeSail: A low cost CubeSat based solar sail demonstration mission," *Advances in Space Research*, vol. 48, no. 2011, pp. 1890-1901, 2011.
- [30] S. Lätt, A. Slavinskis, E. Ilbis, U. Kvell, K. Voormansik, E. Kulu, M. Pajusalu, H. Kuuste and M. Noorma, "ESTCube-1 nanosatellite for electric solar wind sail in-orbit technology demonstration," *Proceedings of the Estonian Academy of Sciences*, vol. 63, no. 2S, pp. 200-209, 2014.

- [31] R. Ridenoure, R. Munakata, A. Diaz, S. Wong, B. Plante, D. Stetson, D. Spencer and J. Foley, "LightSail Program Status: One Down, One to Go," in *29th Annual AIAA/USU Conference on Small Satellites*, Logan, Utah, 2015.
- [32] E. B. Johnson, C. Hardgrove, R. Starr, S. Vogel, R. Frank, G. Stoddard, S. West and J. Christian, "Development of the LunaH-Map miniature neutron spectrometer," in *SPIE Optical Engineering + Applications*, San Diego, California, 2017.
- [33] P. Senatore, A. Klesh, T. H. Zurbuchen, D. McKague and J. Cutler, "Concept, Design, and Prototyping of XSAS: A High Power Extendable Solar Array for CubeSat Applications," in *Proceedings of the 40th Aerospace Mechanisms Symposium*, NASA Kennedy Space Center, Florida, 2010.
- [34] F. Santoni, F. Piergentili, S. Donati, M. Perelli, A. Negri and M. M. Marino, "An Innovative Deployable Solar Panel System for CubeSats," *Acta Astronautica*, vol. 95, no. 2014, pp. 210-217, 2014.
- [35] T. McGuire, M. Hirsch, M. Parsons, S. Leake and J. Straub, "A CubeSat deployable solar panel system," *Proceedings of SPIE*, vol. 9865, no. id 98650C, 2016.
- [36] A. F. Holland, J. Pearson, W. Lysford and J. Straub, "Consideration of the use of origami-style solar panels for use on a terrestrial/orbital wireless power generation and transmission spacecraft," in *SPIE Commercial + Scientific Sensing and Imaging*, Baltimore, Maryland, 2016.
- [37] J. Carr, D. Boyd, A. Martinez, M. SanSoucie, L. Johnson, G. Laue, B. Farmer, J. Smith, B. Robertson and M. Johnson, "The Lightweight Integrated Solar Array and Transceiver (LISA-T): second generation advancements and future of SmallSat power generation," in *30th Annual AIAA/USU Conference on Small Satellites*, AIAA/Utah State University, 2016.
- [38] E. Bashevkin, J. Kenahan, B. Manning, B. Mahlstedt and A. Kalman, "A Novel Hemispherical Anti-Twist Tracking System (HATTS) for CubeSats," in *26th Annual AIAA/USU Conference on Small Satellites*, Logan, Utah, 2012.
- [39] H.-U. Oh and T. Park, "Experimental Feasibility Study of Concentrating Photovoltaic Power System for CubeSat Applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 1942-1949, 2015.
- [40] S. Singh, A. Shrivastav and S. Bhattacharya, "GaN FET based CubeSat Electrical Power System," in *Applied Power Electronics Conference and Exposition (APEC)*, Charlotte, North Carolina, 2015.
- [41] J. Rabinow, "The Magnetic Fluid Clutch," *AIEE Transactions*, vol. 67, no. November, pp. 1308-1315, 1948.
- [42] R. Stanway, "Smart fluids: current and future developments," *Materials Science and Technology*, vol. 20, no. 8, pp. 931-939, 2004.
- [43] C. S. Namuduri, M. A. Golden and J. Praeckel, "Concurrent Research and Development of a Magnetic Ride Control System," in *The 29th Annual Conference of the IEEE*, Roanoke, VA, 2003.

- [44] LORD Corporation, "Automotive Suspension Systems," 2017. [Online]. Available: <http://www.lord.com/products-and-solutions/active-vibration-control/automotive-suspension-systems>. [Accessed 16 March 2017].
- [45] E. J. Park, D. Stoikov, L. F. da Luz and A. Suleman, "A performance evaluation of an automotive magnetorheological brake design with a sliding mode controller," *Mechatronics*, vol. 16, no. 2006, pp. 405-416, 2006.
- [46] E. J. Park, L. F. da Luz and A. Suleman, "Multidisciplinary Design Optimization of an Automotive Magnetorheological Brake Design," *Computers and Structures*, vol. 86, no. 2008, pp. 207-216, 2008.
- [47] K. Karakoc, E. J. Park and A. Suleman, "Design Considerations for an Automotive Magnetorheological Brake," *Mechatronics*, vol. 18, no. 2008, pp. 434-447, 2008.
- [48] L. Yu, L. Ma, J. Song and X. Liu, "Magnetorheological and Wedge Mechanism-Based Brake-by-Wire System with Self-Energizing and Self-Powered Capability by Brake Energy Harvesting," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2568-2580, 2016.
- [49] Design News Editorial, "Brake Cuts Exercise-Equipment Cost," *Design News*, p. 28, 4 December 1995.
- [50] P. Greg, "Exercise Equipment Puts on Magnetorheological Brakes," *Mechanical Engineering*, vol. 119, no. 5, p. 124, 1997.
- [51] C. Stroud, *A Jewel in the Crown: Essays In Honor Of The 75th Anniversary Of The Institute Of Optics*, Rochester, NY: University of Rochester Press, 2004.
- [52] D. Seo and Y. Lee, "Attitude Control Actuator Design with Magnetorheological Fluid Rings," in *26th Annual Space Flight Mechanics Meeting*, Napa, CA, 2016.
- [53] S. Kaluvan, C.-Y. Park and S.-B. Choi, "Bio-Inspired Device: A Novel Smart MR Spring," *Smart Materials and Structures*, p. LT01LT, 2015.
- [54] S. S. Rao, *Mechanical Vibrations*, 5th Edition, Upper Saddle River, NJ: Pearson Prentice Hall, 2011.
- [55] D. Gorinevsky and G. Vukovich, "Nonlinear Input Shaping Control of Flexible Spacecraft Reorientation Maneuver," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 264-270, 1998.
- [56] R. C. Hibbeler, *Structural Analysis*, 7th Edition, Upper Saddle River, NJ: Pearson Prentice Hall, 2009.
- [57] T. Irvine, "Effective Modal Mass & Modal Participation Factors," 11 December 2015. [Online]. Available: <http://www.vibrationdata.com/tutorials2/ModalMass.pdf>. [Accessed 10 April 2018].
- [58] MathWorks Inc., "tfestimate," 2018. [Online]. Available: <https://www.mathworks.com/help/signal/ref/tfestimate.html>.
- [59] MathWorks Inc., "fitmagfrd," 2018. [Online]. Available: <https://www.mathworks.com/help/robust/ref/fitmagfrd.html>.
- [60] MathWorks Inc., "damp," 2018. [Online]. Available: <https://www.mathworks.com/help/control/ref/damp.html>.



- [61] B. K. Kumbhar, S. R. Patil and S. M. Sawant, "Synthesis and Characterization of Magneto-rheological (MR) Fluids for MR Brake Application," *Engineering Science and Technology, an International Journal*, vol. 18, no. 2015, pp. 432-438, 2015.
- [62] L. Johnson, J. Carr, L. Fabisinski and T. R. Lockett, "Lightweight Integrated Solar Array (LISA): Providing Higher Power to Small Spacecraft," in *13th International Energy Conversion Engineering Conference*, Orlando, FL, 2015.
- [63] D. Rankin, D. D. Kekez, R. E. Zee, F. M. Pranajaya, D. G. Foisy and A. M. Beattie, "The CanX-2 NanoSatellite: Expanding the Science Abilities of Nanosatellites," *Acta Astronautica*, vol. 57, no. 2005, pp. 167-174, 2005.
- [64] Westerhoff, et al., "LAICE CubeSat Mission for Gravity Wave Studies," *Advances in Space Research*, vol. 56, no. 2015, pp. 1413-1427, 2015.
- [65] J. Sutherland, "Vibration Damping," 08 September 2004. [Online]. Available: <http://www.mfg.mtu.edu/marc/primers/machtool/vibration/damping.html>.

## Appendix A – Additional Curve-Fitting Plots

The following eight plots comprise the full set of estimated transfer functions and verification plots obtained through MATLAB<sup>™</sup> curve-fitting of the shaker table vibration testing results as described in Section III. C. 2.

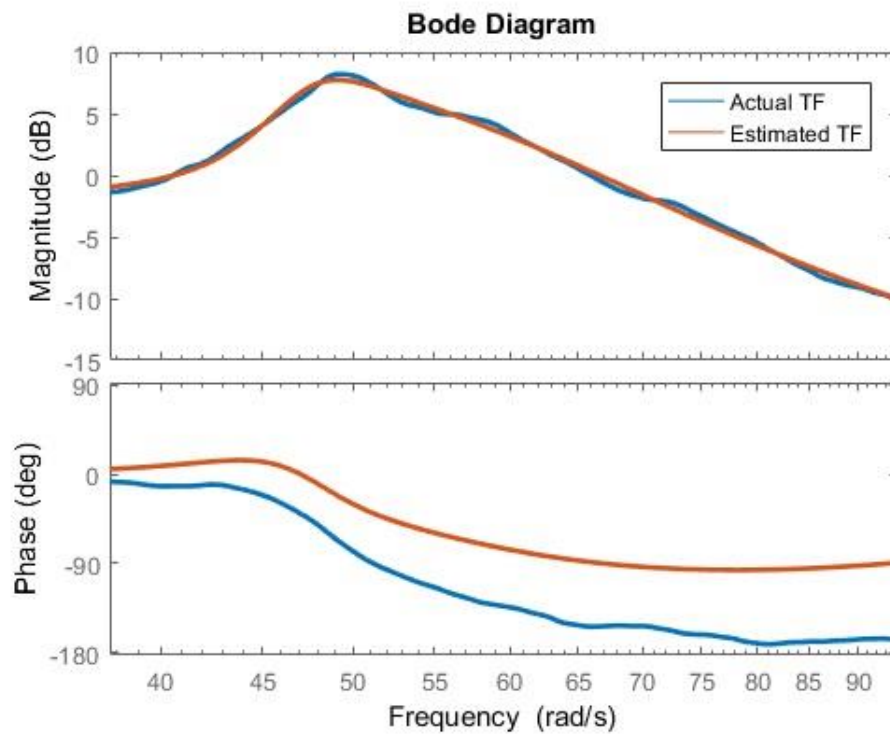


Figure A-1. Bode plot—Estimated versus actual TF in the vicinity of the first mode resonant peak, no field

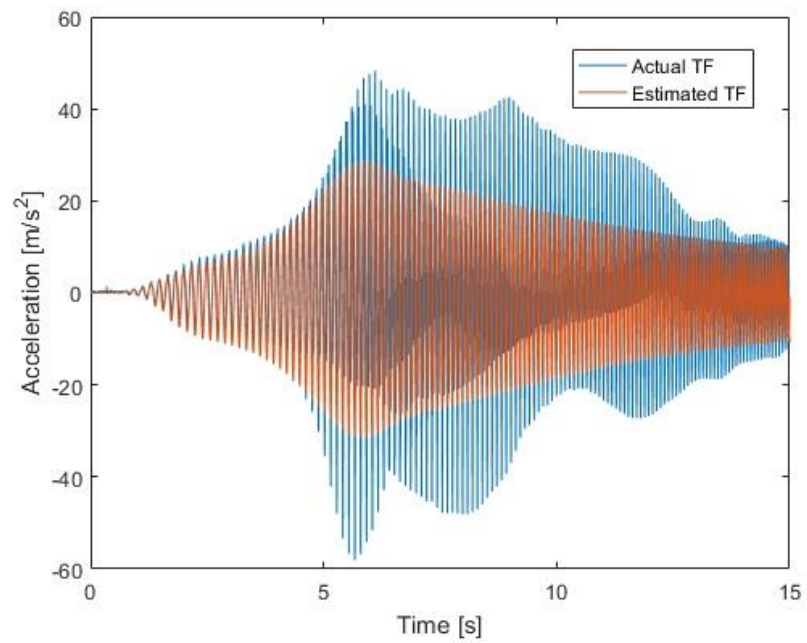


Figure A-2. Actual input signal applied to the estimated first mode transfer function as validation of the estimation

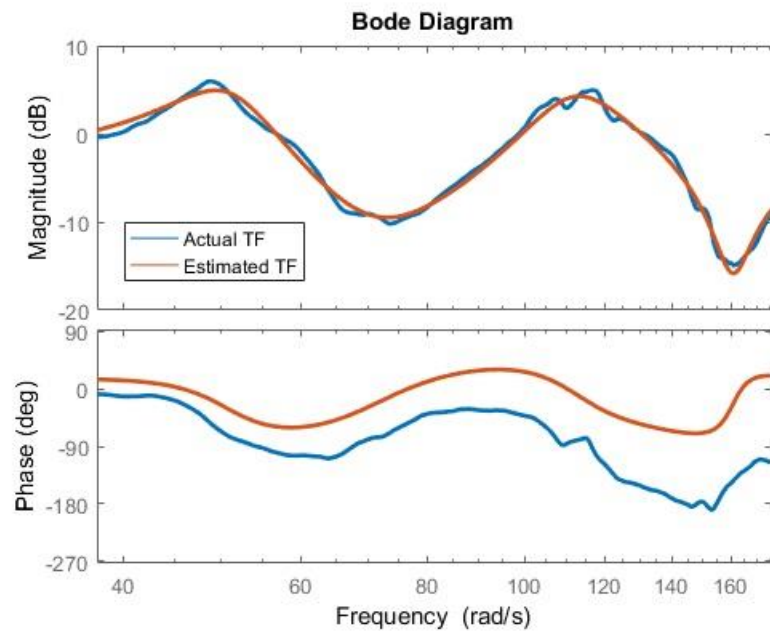


Figure A-3. Bode plot—Estimated versus actual TF in the vicinity of the second mode resonant peak, no field

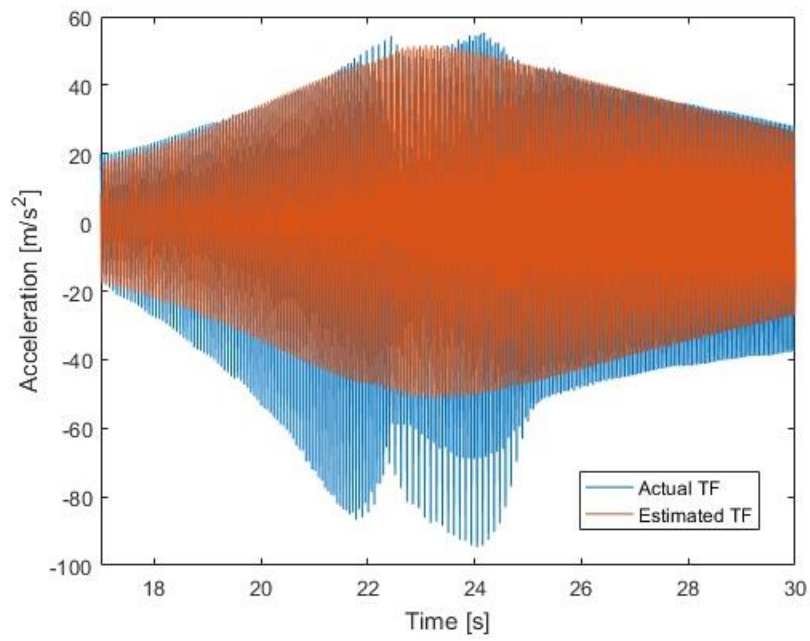


Figure A-4. Actual input signal applied to the estimated second mode transfer function as validation of the estimation

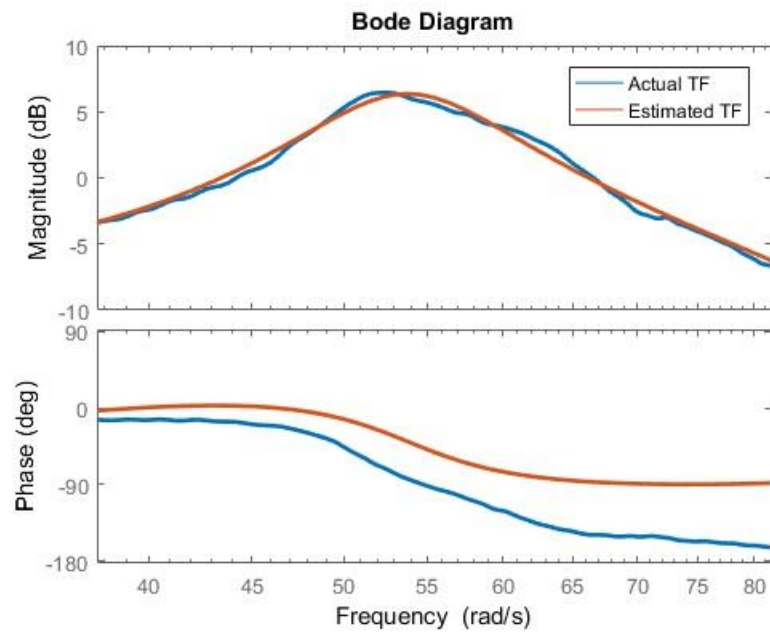


Figure A-5. Bode plot—Estimated versus actual TF in the vicinity of the first mode resonant peak, 25-30mT field

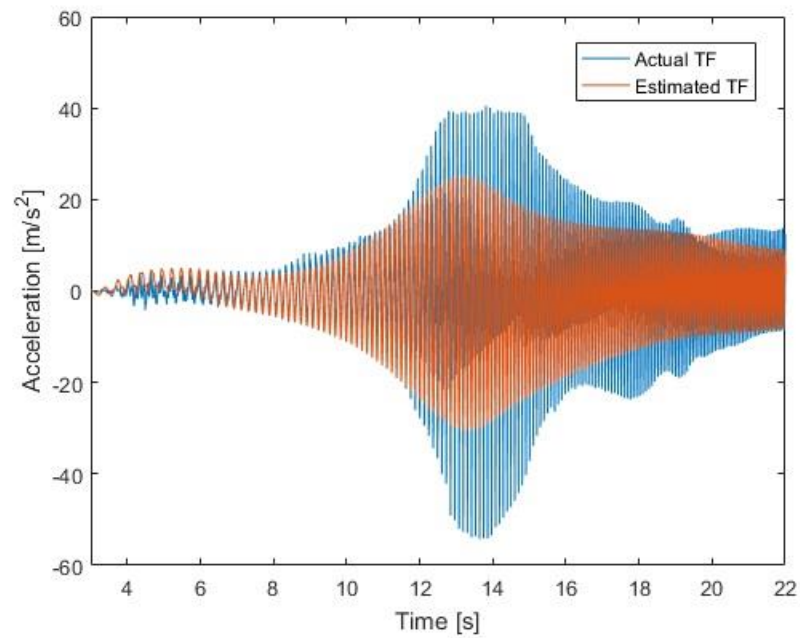


Figure A-6. Actual input signal applied to the estimated first mode transfer function as validation of the estimation

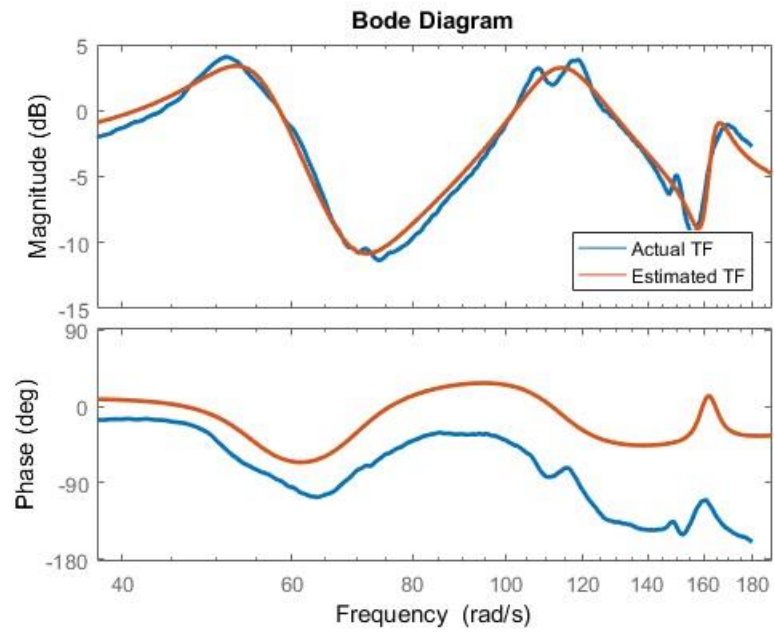


Figure A-7. Bode plot—Estimated versus actual TF in the vicinity of the second mode resonant peak, 25-30mT field

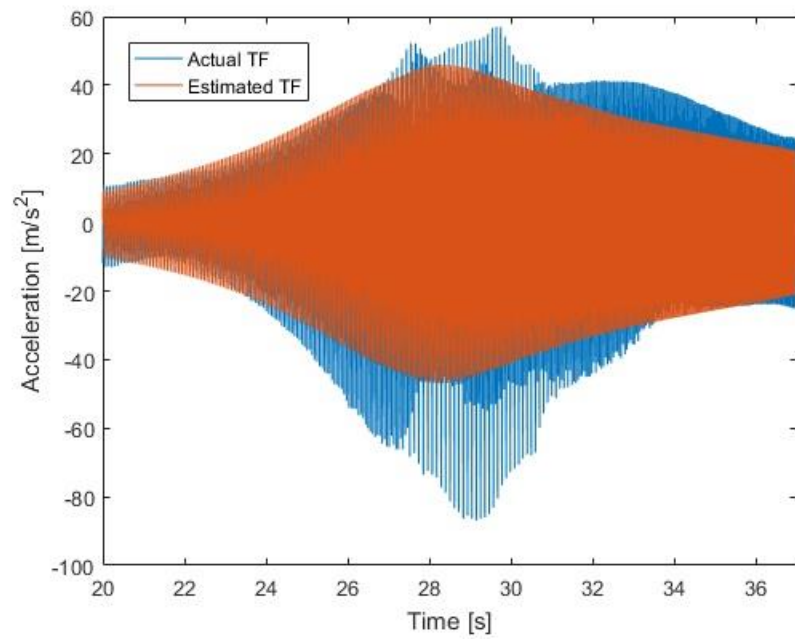


Figure A-8. Actual input signal applied to the estimated second mode transfer function as validation of the estimation

## Appendix B – Numerical Model MATLAB™ Script

MainFunction

```

1      %%HEADER%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2      %Robert Waelchli
3      %waelchrj@gmail.com
4      %April 9, 2018
5      %Embry-Riddle Aeronautical University
6
7      %This simulation models a 3U CubeSat with flexible appendages.
8      %Required Files:
9      %-MainFunction
10     %-SolverFunction
11     %-GainFunction
12     %-PhiFunction
13     %-PlotFunction
14     %-AnimationFunction1
15     %-AnimationFunction2
16     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
17
18     function [ x_Vector ] = MainFunction( dampingEnabled,animationEnabled,...
19         nonlinearEnabled,fieldSatValue )
20
21     %MainFunction is the overall function that runs the simluation.
22     %It can be promoted by the following arguments:
23     %-dampingEnabled      [0]=Disabled [1]=Enabled  [2]=Constant Max Field
24     %-animationEnabled    [0]=Disabled [1]=Enabled
25     %-nonlinearEnabled    [0]=Disabled [1]=Enabled    Linear vs. Non-Linear sim
26     %-fieldSatValue       any positive scalar value   Max Magnetic Field [mT]
27
28     %=====
29     %ELEVATED VISIBILITIES
30     %-----
31     global refVector          %elevated visibility for ODE solver
32     global dampingOption     %elevated visibility for ODE solver
33     global matrixBuilderOption %elevated visibility for ODE solver
34     global nonlinearOption    %elevated visibility for ODE solver
35     global matrixM_qq        %elevated visibility for ODE solver
36     global vectorM           %elevated visibility for ODE solver
37     global IsysX             %elevated visibility for ODE solver
38     global IsysY             %elevated visibility for ODE solver
39     global IsatZ             %elevated visibility for ODE solver
40     global J3                %elevated visibility for ODE solver
41     global sawLength         %elevated visibility for ODE solver
42     global previousTime      %elevated visibility for ODE solver
43     global ei                %elevated visibility for ODE solver
44     global exporter          %elevated visibility for ODE solver
45     global fieldSaturation    %elevated visibility for ODE solver
46     %=====
47
48     %=====
49     %User Defined Constants
50     %-----
51     sawLength = 1;           %SAW length [m]
52     dmpRadius = 0.00635;     %damper radius [m]
53     sawDensity = 2800;       %SAW density--AL is 28k [kg/m^3]
54     dmpDensity = 3000;       %damper density--MRF is ~3k [kg/m^3]
55     sawSpacing = 0.100;     %distance to SAW edge [m]
56     vectorDisp = [2 * pi;pi;0 + pi/2]; %initial displacement [gamma,psi,theta]
57     vectorVel = [0.0;0.0;0]; %initial ang velcty [gamma,psi,theta]

```

```

58 vectorQ = [0;0]; %init modal coord vector [q1,q2]
59 vectorQDot = [0;0]; %init modal deriv vector [q1,q2]
60 refVector = [pi;pi;0]; %ref for lin model [gamma,psi,theta]
61 duration = 20; %duration of simulation [s]
62 frameDelay = 0.05; %animation pause [s]
63 animation = animationEnabled; %[1] = ON, [0] = OFF
64 eValues = [-5; -9; -8;... %eigValues for place function
65 -7; -6; -0.4; -0.5; -1;...
66 -2; -3];
67
68 dampingOption = dampingEnabled; %[0] = Disabled [1] = Enabled
69 matrixBuilderOption = 1; %[1] = Fast & rough [2] = Slow & exact
70 nonlinearOption = nonlinearEnabled; %[0] = linear [1] = nonlinear
71 previousTime = 0; %insantiate the global time variable
72 ei = 0; %instantiate the global summing element
73 fieldSaturation = fieldSatValue; %field saturation value
74 %=====
75
76 %=====
77 %System Properties
78 %-----
79 cubeWidth = 0.100; %hub width [m]
80 cubeHeight = 0.100; %hub height [m]
81 cubeDepth = 0.300; %hub depth [m]
82 sawThickness = 0.002; %SAW thickness [m]
83 cubeMass = 5; %hub mass [kg]
84
85 %=====
86 close all;
87
88 %derived properties from constants and material values
89 sawIzz = ((cubeDepth)*sawThickness^3)/12; %SAW 2nd mmnt area about z-axis
90 sawArea = cubeWidth*sawThickness; %SAW cross-section area
91 sawMu = sawDensity*sawArea; %SAW linear density
92 dmpIzz = pi*(dmpRadius^4)/4; %damper MOI about z-axis
93 dmpArea = pi*dmpRadius^2; %damper cross-section area
94 dmpMu = dmpDensity*dmpArea; %damper linear density
95
96 %build MOI tensors for hub and SAW
97 cubeMOI = CubicMOIBuilderFunction... %hub modeled as cubic shape
98 (cubeMass,cubeWidth,cubeHeight,cubeDepth);
99 sawMOI = CylinderMOIBuilderFunction... %start with SAW as cylinder...
100 ((sawMu * sawLength),0,sawLength );
101 sawMOI = ParallelAxisFunction... %...adjust for off-CG z-axis
102 (sawMOI,3,(sawMu * sawLength),((1/2) * sawLength + sawSpacing));
103 sawMOI = ParallelAxisFunction... %...adjust for off-CG y-axis
104 (sawMOI,2,(sawMu * sawLength),((1/2) * sawLength + sawSpacing));
105 dmpMOI = CylinderMOIBuilderFunction... %start with dmp as cylinder...
106 ((dmpMu * sawLength),dmpRadius,sawLength );
107 dmpMOI = ParallelAxisFunction... %...adjust for off-CG y-axis
108 (dmpMOI,2,(dmpMu * sawLength),((1/2) * sawLength + sawSpacing));
109 dmpMOI = ParallelAxisFunction... %...adjust for off-CG y-axis
110 (dmpMOI,3,(dmpMu * sawLength),((1/2) * sawLength + sawSpacing));
111
112 %=====
113 %in the linearized model, vectorM and matrixM_qq will be static and thus
114 %can be calculated off-line and hardwired in to improve performance:
115 %-----
116 matrixM_qq = [2.9528 0; %calculated previously
117 0 1.5341];
118 vectorM = [1.2330; 0.1418]; %calculated previously
119

```



```

120 %=====
121
122 %=====
123 %derived values
124 %-----
125 J3 = 2 * sawMOI(3,3); %J3 MOI
126 IsatZ = cubeMOI(3,3); %MOI hub about z-axis
127 IsysY = cubeMOI(2,2) + 2 * sawMOI(2,2)... %MOI system about y-axis
128 + 2 * dmpMOI(2,2);
129 IsysX = cubeMOI(1,1) + 2 * sawMOI(1,1)... %MOI system about x-axis
130 + 2 * dmpMOI(1,1);
131
132 %=====
133
134 %initial conditions
135 tspan = [0 duration]; %time vector for use w/ solver
136 x_0 = [vectorDisp(3,1);vectorDisp(2,1);... %initial conditions for solver
137 vectorDisp(1,1);vectorQ(1,1);vectorQ(2,1);...
138 vectorVel(3,1);vectorVel(2,1);vectorVel(1,1);vectorQDot(1,1);...
139 vectorQDot(2,1)];
140
141 [t_Vector,x_Vector] = ode45... %solve ODE numerically w/ode45
142 (@(t,y) SolverFunction(t,y,eValues),tspan,x_0);
143
144 x_Vector(:,(1:3)) = rem(x_Vector... %reduce each element to 0-2pi
145 (:,(1:3)),(2 * pi));
146
147 x_Vector(:,12) = t_Vector;
148
149 previousTime = 0;
150 for m = 1:length(x_Vector) %loop to add derived values
151 ep = x_Vector(m,(4:5)); %ep
152 ed = x_Vector(m,(9:10)); %ed
153
154 dt = t_Vector(m) - previousTime; %time step
155 previousTime = t_Vector(m);
156 ei = (ep * dt) + ei; %ei
157
158 switch dampingOption
159 case 0
160 x_Vector(m,13) = 0; %field strength
161 case 1
162 x_Vector(m,13)... %field strength
163 = GainFunction(ep,ed,ei);
164 case 2
165 x_Vector(m,13)... %field strength
166 = fieldSaturation;
167 end
168 end
169
170 %send results to the plotting function
171 exporter = sort(exporter); %order ODE45 columns
172 PlotFunction(t_Vector,x_Vector);
173
174 %angular displacement vector
175 fprintf('\n\nAngular Displacement Vector\n[%.4f,\t%.4f,\t\t%.4f]', ...
176 x_Vector(end,3),x_Vector(end,2),x_Vector(end,1));
177
178 %angular velocity vector
179 fprintf('\n\nAngular Velocity Vector\n[%.4f,\t%.4f,\t\t%.4f]', ...
180 x_Vector(end,9),x_Vector(end,8),x_Vector(end,7));
181

```

```

182 %modal amplitude vector
183 fprintf('\n\nModal Amplitude Vector\n[%.4e,\t\t%.4e,\t\t%.4e]', ...
184         x_Vector(end,4),x_Vector(end,5),x_Vector(end,6));
185
186 %modal amplitude derivative vector
187 fprintf('\n\nModal Amplitude Derivative Vector');
188 fprintf('\n[%.4e,\t\t%.4e,\t\t%.4e]\n\n',x_Vector(end,10),...
189         x_Vector(end,11),x_Vector(end,12));
190
191 %angular displacement vector in degrees
192 fprintf('\n\nAngular Displacement Vector in Degrees\n');
193 fprintf('[%.4f,\t\t%.4f,\t\t%.4f]',x_Vector(end,3) * 180/pi(),...
194         x_Vector(end,2) * 180/pi(),x_Vector(end,1) * 180/pi());
195
196 %angular velocity vector in degrees
197 fprintf('\n\nAngular Velocity Vector in Degrees-per-Second\n');
198 fprintf('[%.4f,\t\t%.4f,\t\t%.4f]\n\n', x_Vector(end,9) * 180/pi(),...
199         x_Vector(end,8) * 180/pi(),x_Vector(end,7) * 180/pi());
200
201 switch animation
202     case 1
203         %orientation animation
204         AnimationFunction1(t_Vector,x_Vector,frameDelay);
205         %dynamical animation
206         AnimationFunction2(t_Vector,x_Vector,...
207             sawLength,sawSpacing,frameDelay);
208     end
209
210 %%FUNCTIONS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
211 function [ moiTensor ] = ParallelAxisFunction( moiTensor,axis,m,d )
212 %ParallelAxisFunciton accepts a 3x3 MOI tensor, specified axis, mass, and
213 %distance argument and applies parallel axis theroem to the specified axis
214 %(1 = x, 2 = y, 3 = z); it then returns the updated tensor.
215
216 moiTensor(axis,axis) = moiTensor(axis,axis) + (m * d^2);
217
218 end
219
220 function [ moiTensor ] = CubicMOIBuilderFunction( mass,width,height,...
221     depth )
222 %CubicMOIBuilderFunction takes mass, length, width, and depth values and
223 %returns an MOI tensor for a perfect cubic shape.
224
225 moiTensor = zeros(3); %initiate to zeros
226 moiTensor(1,1) = mass * (1/12) * (height^2 + depth^2); %MOI about x
227 moiTensor(2,2) = mass * (1/12) * (width^2 + depth^2); %MOI about y
228 moiTensor(3,3) = mass * (1/12) * (width^2 + height^2); %MOI about z
229 end
230
231 function [ moiTensor ] = CylinderMOIBuilderFunction( mass,radius,...
232     length )
233 %CubicMOIBuilderFunction takes mass, radius, and length values and
234 %returns an MOI tensor for a cylindrical shape.
235
236 moiTensor = zeros(3); %initiate to zeros
237 moiTensor(1,1) = mass * (1/2) * radius^2; %MOI about x
238 moiTensor(2,2) = mass * (1/12) * (length^2); %MOI about y
239 moiTensor(3,3) = mass * (1/12) * (length^2); %MOI about z
240 end
241
242 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
243 end

```

# Solver Function

```

1    function dx_Vector = SolverFunction(t,x,eValues)
2    %dx = 12x1 column vector:
3    %dx(1)  = theta
4    %dx(2)  = psy
5    %dx(3)  = gamma
6    %dx(4)  = q1
7    %dx(5)  = q2
8    %dx(6)  = theta-dot
9    %dx(7)  = psy-dot
10   %dx(8)  = gamma-dot
11   %dx(9)  = q1-dot
12   %dx(10) = q2-dot
13
14   %=====
15   %ELEVATED VISIBILITIES
16   %-----%
17   global refVector           %elevated visibility for ODE solver
18   global dampingOption      %elevated visibility for ODE solver
19   global matrixBuilderOption %elevated visibility for ODE solver
20   global nonlinearOption     %elevated visibility for ODE solver
21   global matrixM_qq         %elevated visibility for ODE solver
22   global vectorM            %elevated visibility for ODE solver
23   global IsysX              %elevated visibility for ODE solver
24   global IsysY              %elevated visibility for ODE solver
25   global IsatZ              %elevated visibility for ODE solver
26   global J3                 %elevated visibility for ODE solver
27   global previousTime       %elevated visibility for ODE solver
28   global ei                 %elevated visibility for ODE solver
29   global fieldSaturation     %elevated visibility for ODE solver
30
31   %=====
32
33   ep = x((4:5),1);           %proportional error signal
34   ed = x((9:10),1);          %derivative error signal
35   dT = t - previousTime;     %time step
36
37   ei = (ep * dT) + ei;
38   previousTime = t;          %update previous time
39
40   switch dampingOption
41       case 0                  %no damping case
42           fieldValue = GainFunction(0,0,0);
43           switch matrixBuilderOption
44               case 1
45                   matrixK_qq... %less exact but much faster
46                       = QuickStiffnessMatrixFunction(0);
47               case 2
48                   matrixK_qq... %time consuming but more exact
49                       = StiffnessMatrixBuilderFunction(0);
50           end
51       case 1                  %normal dynamic damping case
52           fieldValue = GainFunction(ep,ed,ei);
53           switch matrixBuilderOption
54               case 1
55                   matrixK_qq... %less exact but much faster
56                       = QuickStiffnessMatrixFunction(fieldValue);
57               case 2
58                   matrixK_qq... %time consuming but more exact
59                       = StiffnessMatrixBuilderFunction(fieldValue);
60           end

```

```

61         case 2                                %constant field damping case
62         fieldValue = fieldSaturation;
63         switch matrixBuilderOption
64             case 1
65                 matrixK_qq...                    %less exact but much faster
66                 = QuickStiffnessMatrixFunction(fieldValue);
67             case 2
68                 matrixK_qq...                    %time consuming but more exact
69                 = StiffnessMatrixBuilderFunction(fieldValue);
70         end
71     end
72
73     switch dampingOption
74         case 0
75             matrixC_qq = zeros(2);                %system without damping
76         case 1
77             matrixC_qq...                          %damping matrix as a function of field
78             = DampingMatrixBuilderFunction(fieldValue,matrixM_qq,...
79             matrixK_qq);
80         case 2
81             matrixC_qq...                          %constant max damping
82             = DampingMatrixBuilderFunction(fieldSaturation,matrixM_qq,...
83             matrixK_qq);
84     end
85
86     M2 = [(IsatZ + J3...                          %second level M matrix
87           * cos(refVector(2,1))^2) 0 0 (cos(refVector(2,1)) * vectorM')];...
88           0 IsysY 0 0 0; 0 0 IsysX 0 0; (cos(refVector(2,1)) * vectorM)...
89           [0; 0] [0; 0] (2 * matrixM_qq)];
90
91     B2 = zeros(5);                                %initiate 6x6 B2 matrix
92     B2((4:5),(4:5)) = matrixC_qq;                 %second level damping matrix
93
94     K2 = zeros(5);                                %initiate 6x6 K2 to zero
95     K2((4:5),(4:5)) = 2 * matrixK_qq;             %second level K matrix
96
97     D2 = [eye(3);zeros(2,3)];                    %initiate 6x3 D2 to zero
98
99     A = [zeros(5) eye(5);...                      %linear A matrix
100         (- M2\K2) (- M2\B2)];
101     B = [zeros(5,3); (M2\D2)];                   %linear B matrix
102
103     x(1,1) = x(1,1) - refVector(3,1);            %redefine theta as a difrnce from ref
104     x(2,1) = x(2,1) - refVector(2,1);            %redefine psi as a difference from ref
105     x(3,1) = x(3,1) - refVector(1,1);            %redefine gamma as a difrnce from ref
106
107     K = place(A,B,eValues);                       %linear model K matrix
108
109     xCommand = x;                                 %observer can't see q1 and q2
110     xCommand((4:5),1) = 0;
111     xCommand((9:10),1) = 0;
112     u = -K * xCommand;                           %u = -Kx
113
114     exporter = [t u'];                            %create vector element to export
115     assignin('base','varToPassOut',...            %assign to varToPassOut & export to bse
116     exporter);
117     evalin('base',...                             %pass torque values to base variable
118     'varPassedOut((length(varPassedOut) + 1),:) = varToPassOut;');
119
120     switch nonlinearOption
121         case 0
122         %=====

```

```

123 %Implementing Linear Dynamcis
124 %-----%
125 dx_Vector((1:10),1) = (A * x)... %dx = Ax + Bu
126 + (B * u);
127 %=====
128 case 1
129 %=====
130 %Implementing Non-Linear Dynamics
131 %-----%
132 M11 = matrixM_qq(1,1);
133 M12 = matrixM_qq(1,2);
134 M21 = matrixM_qq(2,1);
135 M22 = matrixM_qq(2,2);
136 K11 = matrixK_qq(1,1);
137 K12 = matrixK_qq(1,2);
138 K21 = matrixK_qq(2,1);
139 K22 = matrixK_qq(2,2);
140 B11 = matrixC_qq(1,1);
141 B12 = matrixC_qq(1,2);
142 B21 = matrixC_qq(2,1);
143 B22 = matrixC_qq(2,2);
144
145 dx_Vector(1,1) = x(6);
146 dx_Vector(2,1) = x(7);
147 dx_Vector(3,1) = x(8);
148 dx_Vector(4,1) = x(9);
149 dx_Vector(5,1) = x(10);
150 dx_Vector(6,1) = (2*M11*M22*u(1) - 2*M12*M21*u(1)...
151 - B11*M21*vectorM(2)*x(9)*cos(x(2)) + B11*M22*vectorM(1)*x(9)*cos(x(2))...
152 + B21*M11*vectorM(2)*x(9)*cos(x(2)) - B21*M12*vectorM(1)*x(9)*cos(x(2))...
153 - B12*M21*vectorM(2)*x(10)*cos(x(2)) + B12*M22*vectorM(1)*x(10)*cos(x(2))...
154 + B22*M11*vectorM(2)*x(10)*cos(x(2)) - B22*M12*vectorM(1)*x(10)*cos(x(2))...
155 - 2*K11*M21*vectorM(2)*x(4)*cos(x(2)) + 2*K11*M22*vectorM(1)*x(4)*cos(x(2))...
156 + 2*K21*M11*vectorM(2)*x(4)*cos(x(2)) - 2*K21*M12*vectorM(1)*x(4)*cos(x(2))...
157 - 2*K12*M21*vectorM(2)*x(5)*cos(x(2)) + 2*K12*M22*vectorM(1)*x(5)*cos(x(2))...
158 + 2*K22*M11*vectorM(2)*x(5)*cos(x(2)) - 2*K22*M12*vectorM(1)*x(5)*cos(x(2))...
159 - M11*vectorM(2)^2*x(6)*x(7)*cos(x(2))*sin(x(2)) -
M22*vectorM(1)^2*x(6)*x(7)*cos(x(2))*sin(x(2))...
160 + 2*M11*M22*vectorM(1)*x(7)*x(9)*sin(x(2)) -
2*M12*M21*vectorM(1)*x(7)*x(9)*sin(x(2))...
161 + 2*M11*M22*vectorM(2)*x(7)*x(10)*sin(x(2)) -
2*M12*M21*vectorM(2)*x(7)*x(10)*sin(x(2))...
162 + M12*vectorM(1)*vectorM(2)*x(6)*x(7)*cos(x(2))*sin(x(2))...
163 + M21*vectorM(1)*vectorM(2)*x(6)*x(7)*cos(x(2))*sin(x(2))...
164 + 4*J3*M11*M22*x(6)*x(7)*cos(x(2))*sin(x(2)) - (4*J3*M12*M21*x(6)*x(7)*
cos(x(2)) * sin(x(2)))...
165 /(2*IsatZ*M11*M22 - 2*IsatZ*M12*M21 - M11*vectorM(2)^2*cos(x(2))^2 ...
166 - M22*vectorM(1)^2*cos(x(2))^2 + 2*J3*M11*M22*cos(x(2))^2 ...
167 - 2*J3*M12*M21*cos(x(2))^2 + M12*vectorM(1)*vectorM(2)*cos(x(2))^2 ...
168 + M21*vectorM(1)*vectorM(2)*cos(x(2))^2);
169 dx_Vector(7,1) = -(J3*x(6)^2*cos(x(2))*sin(x(2))...
170 - u(2) + vectorM(1)*x(9)*x(6)*sin(x(2))...
171 + vectorM(2)*x(10)*x(6)*sin(x(2)))/IsysY;
172 dx_Vector(8,1) = u(3)/IsysX;
173 dx_Vector(9,1) = (B11*vectorM(2)^2*x(9)*cos(x(2))^2 +
B12*vectorM(2)^2*x(10)*cos(x(2))^2 ...
174 + 2*K11*vectorM(2)^2*x(4)*cos(x(2))^2 + 2*K12*vectorM(2)^2*x(5)*cos(x(2))^2 ...
175 + 2*M12*u(1)*vectorM(2)*cos(x(2)) - 2*M22*u(1)*vectorM(1)*cos(x(2)) -
2*B11*IsatZ*M22*x(9)...
176 + 2*B21*IsatZ*M12*x(9) - 2*B12*IsatZ*M22*x(10) + 2*B22*IsatZ*M12*x(10)...
177 - 4*IsatZ*K11*M22*x(4) + 4*IsatZ*K21*M12*x(4) - 4*IsatZ*K12*M22*x(5) +
4*IsatZ*K22*M12*x(5)...

```

```

178      - B21*vectorM(1)*vectorM(2)*x(9)*cos(x(2))^2 -
B22*vectorM(1)*vectorM(2)*x(10)*cos(x(2))^2 ...
179      - 2*K21*vectorM(1)*vectorM(2)*x(4)*cos(x(2))^2 -
2*K22*vectorM(1)*vectorM(2)*x(5)*cos(x(2))^2 ...
180      - 2*B11*J3*M22*x(9)*cos(x(2))^2 + 2*B21*J3*M12*x(9)*cos(x(2))^2 ...
181      - 2*B12*J3*M22*x(10)*cos(x(2))^2 + 2*B22*J3*M12*x(10)*cos(x(2))^2 -
4*J3*K11*M22*x(4)*cos(x(2))^2 ...
182      + 4*J3*K21*M12*x(4)*cos(x(2))^2 - 4*J3*K12*M22*x(5)*cos(x(2))^2 ...
183      + 4*J3*K22*M12*x(5)*cos(x(2))^2 +
2*M12*vectorM(2)^2*x(7)*x(10)*cos(x(2))*sin(x(2))...
184      - 2*M22*vectorM(1)^2*x(7)*x(9)*cos(x(2))*sin(x(2)) -
2*IsatZ*M12*vectorM(2)*x(6)*x(7)*sin(x(2))...
185      + 2*IsatZ*M22*vectorM(1)*x(6)*x(7)*sin(x(2))...
186      + 2*M12*vectorM(1)*vectorM(2)*x(7)*x(9)*cos(x(2))*sin(x(2))...
187      - 2*M22*vectorM(1)*vectorM(2)*x(7)*x(10)*cos(x(2))*sin(x(2))...
188      + 2*J3*M12*vectorM(2)*x(6)*x(7)*cos(x(2))^2*sin(x(2))...
189      - 2*J3*M22*vectorM(1)*x(6)*x(7)*cos(x(2))^2*sin(x(2)))/(2*(2*IsatZ*M11*M22 -
2*IsatZ*M12*M21...
190      - M11*vectorM(2)^2*cos(x(2))^2 - M22*vectorM(1)^2*cos(x(2))^2 ...
191      + 2*J3*M11*M22*cos(x(2))^2 - 2*J3*M12*M21*cos(x(2))^2 +
M12*vectorM(1)*vectorM(2)*cos(x(2))^2 ...
192      + M21*vectorM(1)*vectorM(2)*cos(x(2))^2));
193 dx_Vector(10,1) = (B21*vectorM(1)^2*x(9)*cos(x(2))^2 +
B22*vectorM(1)^2*x(10)*cos(x(2))^2 ...
194      + 2*K21*vectorM(1)^2*x(4)*cos(x(2))^2 + 2*K22*vectorM(1)^2*x(5)*cos(x(2))^2 ...
195      - 2*M11*u(1)*vectorM(2)*cos(x(2)) + 2*M21*u(1)*vectorM(1)*cos(x(2)) +
2*B11*IsatZ*M21*x(9)...
196      - 2*B21*IsatZ*M11*x(9) + 2*B12*IsatZ*M21*x(10) - 2*B22*IsatZ*M11*x(10)...
197      + 4*IsatZ*K11*M21*x(4) - 4*IsatZ*K21*M11*x(4) + 4*IsatZ*K12*M21*x(5) -
4*IsatZ*K22*M11*x(5)...
198      - B11*vectorM(1)*vectorM(2)*x(9)*cos(x(2))^2 -
B12*vectorM(1)*vectorM(2)*x(10)*cos(x(2))^2 ...
199      - 2*K11*vectorM(1)*vectorM(2)*x(4)*cos(x(2))^2 -
2*K12*vectorM(1)*vectorM(2)*x(5)*cos(x(2))^2 ...
200      + 2*B11*J3*M21*x(9)*cos(x(2))^2 - 2*B21*J3*M11*x(9)*cos(x(2))^2 +
2*B12*J3*M21*x(10)*cos(x(2))^2 ...
201      - 2*B22*J3*M11*x(10)*cos(x(2))^2 + 4*J3*K11*M21*x(4)*cos(x(2))^2 -
4*J3*K21*M11*x(4)*cos(x(2))^2 ...
202      + 4*J3*K12*M21*x(5)*cos(x(2))^2 - 4*J3*K22*M11*x(5)*cos(x(2))^2 ...
203      - 2*M11*vectorM(2)^2*x(7)*x(10)*cos(x(2))*sin(x(2))...
204      + 2*M21*vectorM(1)^2*x(7)*x(9)*cos(x(2))*sin(x(2)) +
2*IsatZ*M11*vectorM(2)*x(6)*x(7)*sin(x(2))...
205      - 2*IsatZ*M21*vectorM(1)*x(6)*x(7)*sin(x(2))...
206      - 2*M11*vectorM(1)*vectorM(2)*x(7)*x(9)*cos(x(2))*sin(x(2))...
207      + 2*M21*vectorM(1)*vectorM(2)*x(7)*x(10)*cos(x(2))*sin(x(2))...
208      - 2*J3*M11*vectorM(2)*x(6)*x(7)*cos(x(2))^2*sin(x(2))...
209      + 2*J3*M21*vectorM(1)*x(6)*x(7)*cos(x(2))^2*sin(x(2)))/(2*(2*IsatZ*M11*M22 -
2*IsatZ*M12*M21...
210      - M11*vectorM(2)^2*cos(x(2))^2 - M22*vectorM(1)^2*cos(x(2))^2 +
2*J3*M11*M22*cos(x(2))^2 ...
211      - 2*J3*M12*M21*cos(x(2))^2 + M12*vectorM(1)*vectorM(2)*cos(x(2))^2 ...
212      + M21*vectorM(1)*vectorM(2)*cos(x(2))^2));
213
214 %=====
215 end
216
217 %time display to make sure program isn't frozen
218 disp(t);
219
220 end
221
222 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

223
224 %%FUNCTIONS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
225 function [ matrixK_qq ] = QuickStiffnessMatrixFunction( fieldValue )
226 %StiffnessMatrixBuilderFunction accepts arguments for
227 %magnetic field strength. It returns an overall K_qq matrix.
228
229 if (fieldValue >= 0) && (fieldValue < 6)           % < 6mT
230     matrixK_qq = (1.0e+03)...
231     * ([0.0090  -0.0000  -0.0000;
232        -0.0000   0.1833  -0.0002;
233        -0.0000  -0.0002   1.4927]);
234 elseif (fieldValue >= 6) && (fieldValue < 9)       % >= 6mT
235     matrixK_qq = (1.0e+03)...
236     * ([0.0091  -0.0000  -0.0000;
237        -0.0000   0.1859  -0.0002;
238        -0.0000  -0.0002   1.5140]);
239 elseif (fieldValue >= 9) && (fieldValue < 12)      % >= 9mT
240     matrixK_qq = (1.0e+03)...
241     * ([0.0092  -0.0000  -0.0000;
242        -0.0000   0.1885  -0.0002;
243        -0.0000  -0.0002   1.5354]);
244 elseif (fieldValue >= 12) && (fieldValue < 15)    % >= 12mT
245     matrixK_qq = (1.0e+03)...
246     * ([0.0094  -0.0000  -0.0000;
247        -0.0000   0.1911  -0.0002;
248        -0.0000  -0.0002   1.5567]);
249 elseif (fieldValue >= 15) && (fieldValue < 18)    % >= 15mT
250     matrixK_qq = (1.0e+03)...
251     * ([0.0095  -0.0000  -0.0000;
252        -0.0000   0.1937  -0.0002;
253        -0.0000  -0.0002   1.5780]);
254 elseif (fieldValue >= 18) && (fieldValue < 21)    % >= 18mT
255     matrixK_qq = (1.0e+03)...
256     * ([0.0096  -0.0000  -0.0000;
257        -0.0000   0.1964  -0.0002;
258        -0.0000  -0.0002   1.5994]);
259 elseif (fieldValue >= 21) && (fieldValue < 24)    % >= 21mT
260     matrixK_qq = (1.0e+03)...
261     * ([0.0098  -0.0000  -0.0001;
262        -0.0000   0.1990  -0.0002;
263        -0.0001  -0.0002   1.6207]);
264 elseif (fieldValue >= 24) && (fieldValue < 27)    % >= 24mT
265     matrixK_qq = (1.0e+03)...
266     * ([0.0099  -0.0000  -0.0001;
267        -0.0000   0.2016  -0.0002;
268        -0.0001  -0.0002   1.6421]);
269 elseif (fieldValue >= 27) && (fieldValue < 30)    % >= 27mT
270     matrixK_qq = (1.0e+03)...
271     * ([0.0100  -0.0000  -0.0001;
272        -0.0000   0.2042  -0.0002;
273        -0.0001  -0.0002   1.6634]);
274 elseif (fieldValue >= 30) && (fieldValue < 33)    % >= 30mT
275     matrixK_qq = (1.0e+03)...
276     * ([0.0101  -0.0000  -0.0001;
277        -0.0000   0.2068  -0.0002;
278        -0.0001  -0.0002   1.6848]);
279 elseif (fieldValue >= 33) && (fieldValue < 36)    % >= 33mT
280     matrixK_qq = (1.0e+03)...
281     * ([0.0103  -0.0000  -0.0001;
282        -0.0000   0.2095  -0.0002;
283        -0.0001  -0.0002   1.7061]);
284 elseif (fieldValue >= 36) && (fieldValue < 39)    % >= 36mT

```

```

285     matrixK_qq = (1.0e+03)...
286     * ([0.0104   -0.0000   -0.0001;
287        -0.0000    0.2121   -0.0002;
288        -0.0001   -0.0002    1.7275]);
289 elseif (fieldValue >= 39) && (fieldValue < 42) % >= 39mT
290     matrixK_qq = (1.0e+03)...
291     * ([0.0105   -0.0000   -0.0001;
292        -0.0000    0.2147   -0.0002;
293        -0.0001   -0.0002    1.7488]);
294 elseif (fieldValue >= 42) && (fieldValue < 45) % >= 42mT
295     matrixK_qq = (1.0e+03)...
296     * ([0.0107   -0.0000   -0.0001;
297        -0.0000    0.2173   -0.0002;
298        -0.0001   -0.0002    1.7702]);
299 elseif (fieldValue >= 45) && (fieldValue < 48) % >= 45mT
300     matrixK_qq = (1.0e+03)...
301     * ([0.0108   -0.0000   -0.0001;
302        -0.0000    0.2199   -0.0003;
303        -0.0001   -0.0003    1.7915]);
304 elseif (fieldValue >= 48) && (fieldValue < 51) % >= 48mT
305     matrixK_qq = (1.0e+03)...
306     * ([0.0109   -0.0000   -0.0001;
307        -0.0000    0.2226   -0.0003;
308        -0.0001   -0.0003    1.8128]);
309 elseif (fieldValue >= 51) && (fieldValue < 54) % >= 51mT
310     matrixK_qq = (1.0e+03)...
311     * ([0.0110   -0.0000   -0.0001;
312        -0.0000    0.2252   -0.0003;
313        -0.0001   -0.0003    1.8342]);
314 elseif (fieldValue >= 54) && (fieldValue < 57) % >= 54mT
315     matrixK_qq = (1.0e+03)...
316     * ([0.0112   -0.0000   -0.0001;
317        -0.0000    0.2278   -0.0003;
318        -0.0001   -0.0003    1.8555]);
319 elseif (fieldValue >= 57) && (fieldValue < 60) % >= 57mT
320     matrixK_qq = (1.0e+03)...
321     * ([0.0113   -0.0000   -0.0001;
322        -0.0000    0.2304   -0.0003;
323        -0.0001   -0.0003    1.8769]);
324 else % >= 60mT
325     matrixK_qq = (1.0e+03)...
326     * ([0.0114   -0.0000   -0.0001;
327        -0.0000    0.2331   -0.0003;
328        -0.0001   -0.0003    1.8982]);
329 end
330
331 matrixK_qq = matrixK_qq(1:2,1:2); %only the first 2x2
332
333 end
334
335 function [ matrixK_qq ] = StiffnessMatrixBuilderFunction...
336     ( fieldValue )
337 %StiffnessMatrixBuilderFunction accepts arguments for
338 %magnetic field strength. It returns an overall K_qq matrix.
339
340 %=====
341 %Elevated Visibilities
342 %-----
343 global sawLength %elevated visibility for solver
344
345 %=====
346

```



```

347 EI = ((0.028/30) * fieldValue) + 0.193;           %based off SCI-Tech ppr
348
349 %define domain array
350 domain_x = 0:(sawLength/10000):sawLength;
351
352 %initiate the modal stiffness matrix
353 matrixK_qq = zeros(3);                             %initialize 3x3 matrix
354
355 %populate the stiffness matrix
356 for m = 1:3
357     for n = 1:3
358         y_m = PhiFunction(domain_x,m,...           %def range for m counter
359                             sawLength);
360         y_n = PhiFunction(domain_x,n,...           %def range for n counter
361                             sawLength);
362         stepSize = (sawLength/10000);              %step size for num. diff.
363         y_mDoublePrime = (diff(diff(y_m)...        %numerical double diff. m
364                                 /stepSize)...
365                                 /stepSize);
366         y_nDoublePrime = (diff(diff(y_n)...        %numerical double diff. n
367                                 /stepSize)...
368                                 /stepSize);
369         y = (y_mDoublePrime)...                    %phi_m" * phi_n" range
370             * diag(y_nDoublePrime);
371
372         %in order to integrate, MATLAB requires x be equal
373         %to y, which is no longer the case post double-differentiation so
374         %a new domain must be created of equal length to the range
375         domain_x_2 = 0:(sawLength/(length(domain_x)-3)):sawLength;
376
377         %numerically integrate for solution
378         matrixK_qq(m,n) = 2 * (EI) * trapz(domain_x_2,y);
379     end
380 end
381
382 end
383
384 function [ matrixC_qq ] = DampingMatrixBuilderFunction(fieldValue,...
385     matrixM_qq,matrixK_qq)
386 %this function accepts an argument for magnetic field strength
387 %and returns a proportional damping matrix
388
389 zeta1 = ((0.033/30) * fieldValue)...              %first mode damping ratio
390         + 0.060;
391 zeta2 = 0.098;                                     %second mode damping ratio
392
393 crit1 = 2 * sqrt(matrixM_qq(1,1))...               %first mode critical damping
394         * sqrt(matrixK_qq(1,1));
395 crit2 = 2 * sqrt(matrixM_qq(2,2))...               %second mode critical damping
396         * sqrt(matrixK_qq(2,2));
397
398 matrixC_qq = zeros(2);
399 matrixC_qq(1,1) = zeta1 * crit1;                   %first mode damping coefficient
400 matrixC_qq(2,2) = zeta2 * crit2;                   %second mode damping coefficient
401
402 end
403
404 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## GainFunction

```

1    function temp = GainFunction(ep,ed,ei)
2    %this function accepts three arguments for proportional, derivative, and
3    %integral error signals, applies appropriate gains, and returns a final
4    %signal for the field value
5
6    %=====
7    %ELEVATED VISIBILITIES
8    %-----
9    global kp                                %elevated visibility for solver
10   global kd                                %elevated visibility for solver
11   global ki                                %elevated visibility for solver
12   global fieldSaturation                    %elevated visibility for solver
13   global dampingOption                      %elevated visibility for solver
14
15   %=====
16
17   %=====
18   %User Defined Constants
19   %-----
20   saturationLimit = fieldSaturation; %limit the max strength of mag field
21
22   %=====
23
24   kp = 1e4;                                %proportional gain--accept at 1e3
25   kd = 1e3;                                %derivative gain--accept at 1e2
26   ki = 1e2;                                %integral gain--accept at 1e2
27
28   % temp = norm(ep * kp) + norm(ed * kd) + norm(ei * ki);
29   temp = norm((ep * kp)+(ed * kd)+(ei * ki));
30
31   if temp > fieldSaturation                    %conditional overrides
32       temp = saturationLimit;                %field saturation
33   elseif dampingOption == 2
34       temp = fieldSaturation;                %max case example
35   end
36
37   end

```

## PhiFunction

```

1    function [ temp ] = PhiFunction( domain,noMode,length )
2    %function takes in an x domain and a mode number and returns the range
3    %vector for a cantilever beam
4
5    Bn_Vector = [1.875104 4.694091...          %define Bn_L cases
6                7.854757];
7    Bn_L = Bn_Vector(noMode);                %assign Bn_L value for trig
8    Bn_x = Bn_L * (domain/length);           %define Bn_x value for trig
9
10   %define alpha_n constant based on Bn_L value
11   alpha_n = (sin(Bn_L) + sinh(Bn_L))/(cos(Bn_L) + cosh(Bn_L));
12
13   %solve for range vector and return
14   temp = (sin(Bn_x) - sinh(Bn_x)...
15         - alpha_n * (cos(Bn_x) - cosh(Bn_x)));
16   end

```

# PlotFunction

```

1    function PlotFunction( t_Vector,x_Vector )
2    %%HEADER%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3    %This function is part of a larger program to calculate the dynamic
4    %response of a 3U CubeSat to excitation force and the free-response
5    %of the CubeSat while a MRF damper controls the reaction.
6
7    %Robert Waelchli
8    %April 9, 2018
9
10   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12   %%SCRIPT%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13   %=====
14   %Plotting Format Options
15   %-----%
16   lineWidth = 1.5;
17   fontSize = 12;
18
19   %=====
20
21   %build and display a three plot figure for roll, pitch, and yaw rates
22   f1 = figure('Name','Angles','Color',...    %create figure and make white
23       [1 1 1]);
24   hold on
25   subplot(3,1,1);                          %place the roll-rate subplot
26   plot(t_Vector,x_Vector(:,3),'LineWidth',... %create roll-rate subplot
27       lineWidth);set(gca,'fontSize', fontSize);
28   ylabel('\gamma (rad)', 'FontSize',...      %y-axis label
29       fontSize, 'FontWeight', 'bold');
30
31   figure(f1);
32   subplot(3,1,2);                          %place the pitch-rate subplot
33   plot(t_Vector,x_Vector(:,2),'LineWidth',... %create pitch-rate subplot
34       lineWidth);set(gca,'fontSize', fontSize);
35   ylabel('\psi (rad)', 'FontSize',...        %y-axis label
36       fontSize, 'FontWeight', 'bold');
37
38   figure(f1);
39   subplot(3,1,3);                          %place the yaw-rate subplot
40   plot(t_Vector,x_Vector(:,1),'LineWidth',... %create yaw-rate plot
41       lineWidth);set(gca,'fontSize', fontSize);
42   xlabel('Time (s)', 'FontSize', fontSize,... %x-axis label
43       'FontWeight', 'bold');
44   ylabel('\theta (rad)', 'FontSize',...      %y-axis label
45       fontSize, 'FontWeight', 'bold');
46   hold off;
47
48   % %build and display a three plot figure for the 2 modal-coordinates
49   f2 = figure('Name','Modal Amplitudes',... %create figure and make white
50       'Color',[1 1 1]);
51   hold on
52   subplot(3,1,1);                          %place the q1 subplot
53   plot(t_Vector,x_Vector(:,4),'LineWidth',... %create q1 subplot
54       lineWidth);set(gca,'fontSize', fontSize);
55   ylabel('q1 amplitude', 'FontSize',...      %y-axis label
56       fontSize, 'FontWeight', 'bold');
57
58   figure(f2);
59   subplot(3,1,2);                          %place the q2 subplot
60   plot(t_Vector,x_Vector(:,5),'LineWidth',... %create q2 subplot

```

```

61     lineWidth);set(gca,'fontSize', fontSize);
62     ylabel('q2 amplitude', 'FontSize',... %y-axis label
63     fontSize, 'FontWeight', 'bold');
64
65     figure(f2);
66     subplot(3,1,3); %field strength subplot
67     plot(t_Vector,x_Vector(:,13),'LineWidth',...
68     lineWidth);set(gca,'fontSize', fontSize);
69     xlabel('Time (s)', 'FontSize', fontSize,... %x-axis label
70     'FontWeight', 'bold');
71     ylabel('Field Strength (mT)', 'FontSize',... %y-axis label
72     fontSize, 'FontWeight', 'bold');
73
74     hold off;
75
76     end
77
78     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

#### AnimationFunction1

```

1     function AnimationFunction1( t,x,frameDelay )
2     %AnimationFunction accepts a time and state vector and animates the motion
3
4     tmax = max(t);
5     newT = (0:(1/20):tmax).';
6
7     f3 = figure('Name',... %create new figure
8     'Rotation Animation','Color',[1 1 1]);
9
10    xlabel('LVLH X','FontWeight','bold');
11    ylabel('LVLH Y','FontWeight','bold');
12    zlabel('LVLH Z','FontWeight','bold');
13
14    O = [0,0,0]; %origin
15    axis (gca, 'equal'); %sets aspect ratio
16    axis ([ -1.5 1.5 -1.5 1.5 -1.5 1.5 ]); %sets plot limits
17    grid on; %turns on grid
18
19    %loop for animation
20    for m = 1:length(newT)
21        [~,addy] = min(abs(newT(m) - t)); %closes match to desired time index
22
23        dcm = angle2dcm(x(addy,1),... %build cosine matrix
24        x(addy,2),x(addy,3), 'ZYX');
25        xBFF = dcm(:,1); %x coordinate is first column
26        yBFF = dcm(:,2); %y coordinate is second column
27        zBFF = dcm(:,3); %z coordinate is third column
28
29        ptsX = [0;transp(xBFF)]; %matrix of 0-vector and transp x-pt
30        ptsY = [0;transp(yBFF)]; %matrix of 0-vector and transp y-pt
31        ptsZ = [0;transp(zBFF)]; %matrix of 0-vector and transp z-pt
32
33        figure(f3);
34        xLine = line(ptsX(:,1),... %clmns of ptsX are crdnts for xline
35        ptsX(:,2), ptsX(:,3),'LineWidth',2);
36        yLine = line(ptsY(:,1),... %clmns of ptsY are crdnts for yline
37        ptsY(:,2), ptsY(:,3),'LineWidth',2);
38        zLine = line(ptsZ(:,1),... %clmns of ptsZ are crdnts for zline
39        ptsZ(:,2), ptsZ(:,3),'LineWidth',2);
40

```

```

41     xText = text(xBFF(1,1) + 0.1,...    %display 'x' label on axis
42                xBFF(2,1) + 0.1,xBFF(3,1),'X');
43     yText = text(yBFF(1,1) + 0.1,...    %display 'y' label on axis
44                yBFF(2,1) + 0.1,yBFF(3,1),'Y');
45     zText = text(zBFF(1,1),...         %display 'z' label on axis
46                zBFF(2,1), zBFF(3,1) + 0.1,'Z');
47     timeText = text(-1,-1,-1,...       %display current time on plot
48                sprintf('Time: %0.3f',t(addy)));
49
50     if m == 1
51         pause(1);                      %pause for 5 seconds for recording
52     elseif m == length(newT)
53         pause(1);
54     else
55         pause(frameDelay);              %pause a tic so the user can see
56     end
57
58     if addy < length(t)                 %if not at the last item...
59         delete(xLine);                  %...delete the xaxis
60         delete(yLine);                  %...delete the yaxis
61         delete(zLine);                  %...delete the zaxis
62
63         delete(xText);                  %...delete the xaxis label
64         delete(yText);                  %...delete the yaxis label
65         delete(zText);                  %...delete the zaxis label
66         delete(timeText);               %...delete the time
67     end
68 end
69
70 end

```

#### AnimationFunction2

```

1     function AnimationFunction2( t,x,sawLength,spacing,frameDelay )
2     %AnimationFunction accepts a time, a state vector, and some vehicle
3     %vehicle parameters and then animates the motion
4
5     tmax = max(t);
6     newT = (0:(1/15):tmax).';
7
8     %create an x domain from one end of the solar array to the other
9     domain_x1 = transp...               %1st full length domain for functn
10    ( - (sawLength + spacing):2 * (sawLength + spacing)/200:...
11    (sawLength + spacing));
12    domain_x2 = domain_x1;               %2nd full length domain for signs
13    domain_x3 = domain_x1;               %3rd full length domain for plotting
14    domain_x2(domain_x2 < 0) = -1;        %use 2nd domain to create sign vctr
15    domain_x2(domain_x2 >= 0) = 1;        %use 2nd domain to create sign vctr
16    domain_x1 = abs(domain_x1);           %make 1st domain all positive
17    domain_x1 = domain_x1 - spacing;      %shift all values for spacing
18    domain_x1(abs(domain_x1) <= spacing)... %set center values to zero for hub
19    = 0;
20
21    f4 = figure(4);                      %create new figure
22
23    %loop for animation
24    for m = 1:length(newT)
25        [~,addy] = min(abs(newT(m) - t)); %closes match to desired time index
26
27        range_y = zeros...               %instance a range vector at zeros
28        (length(domain_x1),1);

```

```

29
30     %loop for range
31     for n = 1:2
32         range_y(:,1) = range_y...           %sum first 2 mode shapes
33         + (x(addy,n + 3) * (domain_x2...
34         .* PhiFunction(domain_x1,n,sawLength)));
35     end
36
37     figure(f4);
38     sawShape = plot(domain_x3,...           %plot y vs x
39         range_y,'linewidth',2);
40     %     axis ([ -1.5 1.5...               %sets plot limits
41     %         -0.25 0.25 ]);
42     axis ([ -1.5 1.5...                   %sets plot limits
43         -0.3 0.3 ]);
44     grid on;
45     timeText = text(-1.0, -0.2,...         %display current time on plot
46         sprintf('Time: %0.3f',t(addy)));
47     xlabel('BFF X [m]',...                %aesthetic choice
48         'FontWeight','bold');
49     ylabel('BFF Y [m]',...                %aesthetic choice
50         'FontWeight','bold');
51
52     if m == 1
53         pause(1);                         %pause for 5 seconds for recording
54     elseif m == length(newT)
55         pause(1);                         %pause for 5 seconds for recording
56     else
57         pause(frameDelay);                %pause a tic so the user can see
58     end
59
60     if addy < length(t)                   %if not at the last item...
61         delete(sawShape);                 %...delete the plot
62         delete(timeText);                 %...delete the time
63     end
64 end
65
66 end
67
68 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```